

# Novell eDirectory™

8.7.3

管理ガイド

[www.novell.com](http://www.novell.com)

2004年1月9日



Novell®

## 法令通知

米国 Novell, Inc. およびノベル株式会社は、本書の内容または本書を使用した結果について、いかなる保証、表明または約束も行っておりません。また、本書の商品性、および特定の用途への適合性について、いかなる黙示的保証も否認し、排除します。また、本書の内容は予告なく変更されることがあります。

米国 Novell, Inc. およびノベル株式会社は、すべてのノベル製ソフトウェアについて、いかなる保証、表明または約束も行っておりません。また、ノベル製ソフトウェアの商品性、および特定の用途への適合性について、いかなる黙示的保証も否認し、排除します。米国 Novell, Inc. およびノベル株式会社は、ノベル製ソフトウェアの内容を変更する権利を常に留保します。

本製品を、米国の輸出規制およびユーザの居住する国の法律に違反して輸出する、または再輸出することはできません。

Copyright © 2002-2003 Novell, Inc. All rights reserved. 本書の一部または全体を無断で複写・転載することは、その形態を問わず禁じます。

米国特許番号 5,608,903; 5,671,414; 5,677,851; 5,758,344; 5,784,560; 5,794,232; 5,818,936; 5,832,275; 5,832,483; 5,832,487; 5,870,739; 5,873,079; 5,878,415; 5,884,304; 5,913,025; 5,919,257; 5,933,826. 米国および国際特許未決定。

Novell, Inc.  
1800 South Novell Place  
Provo, UT 84606  
U.S.A.

[www.novell.com](http://www.novell.com)

Novell eDirectory 8.7.3 管理ガイド  
2004 年 1 月 9 日

**オンラインドキュメント**：本製品およびその他の Novell 製品のオンラインマニュアルにアクセスする場合や、アップデート版を取得する場合は、<http://www.novell.com/documentation/japanese> を参照してください。

## Novell の商標

ConsoleOne は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

DirXML は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

eDirectory は、米国 Novell, Inc. の商標です。

Internetwork Packet Exchange および IPX は、米国 Novell, Inc. の商標です。

NetWare は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

NetWare Core Protocol および NCP は、米国 Novell, Inc. の商標です。

NetWare Loadable Module および NLM は、米国 Novell, Inc. の商標です。

NetWare SMS は、米国 Novell, Inc. の商標です。

NMAS は、米国 Novell, Inc. の商標です。

Novell は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

Novell Certificate Server は、米国 Novell, Inc. の商標です。

Novell Client は、米国 Novell, Inc. の商標です。

Novell Directory Services および NDS は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

Novell SecretStore は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

Transaction Tracking System および TTS は、米国 Novell, Inc. の商標です。

ZENworks は、米国 Novell, Inc. の米国ならびに他の国々における登録商標です。

## サードパーティの商標

サードパーティ各社とその製品の商標は、所有者であるそれぞれの会社に所属します。

この製品には、OpenSSL プロジェクトが開発した OpenSSL Toolkit(<http://www.openssl.org>) で使用するソフトウェアが含まれています。



# 目次

このガイドについて	11
<b>1 Novell eDirectory について</b>	<b>13</b>
新機能	14
Novell eDirectory	14
Novell iManager の便利な管理機能	15
強力なツリー構造	16
Web ベースの管理ユーティリティ	18
シングルログインと認証	19
オブジェクトクラスとプロパティ	19
オブジェクトのリスト	19
コンテナオブジェクトクラス	22
リーフオブジェクトクラス	25
コンテキストと命名規則	35
識別名	36
タイプ付きの名前	36
ネームレゾリューション	36
現在のワークステーションのコンテキスト	37
先頭ピリオド	37
相対命名	37
後続ピリオド	37
UNIX でのコンテキストと命名規則	38
スキーマ	38
スキーマ管理	39
スキーマクラス、属性、および構文	39
必須属性およびオプション属性について	43
スキーマの例	44
スキーマを設計する	44
パーティション	44
パーティション	45
パフォーマンス向上のためにレプリカを分散する	46
パーティションと WAN リンク	46
レプリカ	48
レプリカのタイプ	49
フィルタ済みレプリカ	51
NetWare バインダリエミュレーション	52
レプリカリングでのサーバの同期	53
リソースへのアクセス	53
eDirectory での権利	54
トラスティ割り当ておよびターゲット	54
eDirectory での権利の概念	54
新規サーバのデフォルト権	59
管理の委託	60
権利の管理	61

<b>2</b>	<b>Novell eDirectory ネットワークの設計</b>	<b>67</b>
	eDirectory 設計の基本	67
	ネットワークのレイアウト	67
	組織の構造	68
	eDirectory 設計の準備をする	68
	eDirectory ツリーの設計	68
	命名標準ドキュメントを作成する	68
	Tree の上位層を設計する	71
	ツリーの下位層を設計する	73
	ツリーのパーティション化のガイドライン	74
	ツリーの上位層におけるパーティションを決定する	74
	ツリーの下位層におけるパーティションを決定する	75
	パーティションサイズを決定する	75
	ネットワーク変数について	76
	ツリーのレプリカ作成に関するガイドライン	76
	ワークグループのニーズ	76
	障害対策	76
	レプリカ数を決定する	77
	ツリーパーティションのレプリカを作成する	77
	管理用にレプリカを作成する	78
	NetWare のバインダリサービスの必要性に対応させる	78
	WAN トラフィックを管理する	78
	ユーザ環境についてのプランニング	79
	ユーザの必要条件を検討する	79
	アクセスに関するガイドラインを作成する	79
	e ビジネスに対応する eDirectory の設計	79
	Novell Certificate Server について	80
	Novell Certificate Server でタスクを実行するのに必要な権利	81
	Linux、Solaris、AIX、および HP-UX システムでの eDirectory 操作に関するセキュリティを確保する	82
	ネットワーク時刻の同期	85
	NetWare サーバで時刻を同期する	85
	Windows サーバで時刻を同期する	86
	Linux、Solaris、AIX、または HP-UX システムで時刻を同期する	86
	時刻同期を確認する	87
<b>3</b>	<b>オブジェクトの管理</b>	<b>89</b>
	オブジェクトに関連する一般的なタスク	89
	eDirectory ツリーを参照する	90
	オブジェクトを作成する	92
	オブジェクトのプロパティを変更する	92
	オブジェクトをコピーする	93
	オブジェクトを移動する	93
	オブジェクトを削除する	93
	オブジェクトをリネームする	94
	ユーザアカウントを管理する	94
	ユーザアカウントを作成および変更する	94
	オプションのアカウント機能を設定する	95
	ログインスクリプトを設定する	97
	リモートユーザのログイン時間制限	99
	ユーザアカウントを削除する	99
	役割ベースサービスを設定する	100
	RBS 役割を定義する	102
	カスタム RBS タスクを定義する	103

<b>4</b>	<b>スキーマの管理</b>	<b>105</b>
	スキーマの拡張	106
	クラスを作成する	106
	クラスを削除する	106
	属性を作成する	107
	クラスへオプション属性を追加する	107
	属性を削除する	107
	補助クラスを作成する	108
	補助クラスのプロパティでオブジェクトを拡張する	108
	オブジェクトの補助プロパティを変更する	108
	オブジェクトから補助プロパティを削除する	109
	スキーマの表示	109
	クラス情報を参照する	109
	属性情報を表示する	110
	手動でスキーマを拡張する	110
	NetWare でスキーマを拡張する	110
	Windows でスキーマを拡張する	110
	Linux、Solaris、AIX、または HP-UX システムでスキーマを拡張する	111
	eDirectory 8.7 に追加されたスキーマフラグ	112
	eMBox クライアントを使用してスキーマ操作を実行する	114
	DSSchema eMTool を使用する	114
	DSSchema eMTool オプション	115
<b>5</b>	<b>パーティションおよびレプリカの管理</b>	<b>117</b>
	パーティションの作成	118
	パーティションのマージ	119
	パーティションの移動	120
	パーティションの作成操作またはマージ操作のキャンセル	121
	レプリカの管理	121
	レプリカを追加する	122
	レプリカを削除する	122
	レプリカタイプを変更する	123
	フィルタ済みレプリカを設定し管理する	125
	フィルタ処理済レプリカウィザードを使用する	125
	パーティションスコープを定義する	126
	サーバフィルタを設定する	126
	パーティションおよびレプリカを表示する	127
	サーバのパーティションを表示する	128
	パーティションレプリカを表示する	128
	パーティションに関する情報を表示する	128
	パーティションの階層を表示する	128
	レプリカに関する情報を表示する	128
<b>6</b>	<b>Novell eDirectory 管理ユーティリティ</b>	<b>131</b>
	Novell インポート / エクスポート変換ユーティリティ	131
	Novell iManager インポート / エクスポート変換ウィザードを使用する	132
	コマンドラインインタフェースを使用する	135
	変換ルール	151
	LBURP(LDAP Bulk Update/Replication Protocol)	160
	LDAP ディレクトリ間でスキーマを移行する	161
	LDIF のインポートを高速化する	161
	インデックスマネージャ	163
	インデックスを作成する	164
	インデックスを削除する	165
	インデックスをオフラインにする	165
	他のサーバ上でインデックスを管理する	165

Novell インポート / エクスポート変換ユーティリティを使用してインデックスを管理する	166
プレディケートデータ	168
プレディケートデータを管理する	168
eDirectory Service Manager	168
eMBox クライアントのサービスマネージャ eMTool を使用する	169
Novell iManager でサービスマネージャプラグインを使用する	170
<b>7 Novell iMonitor 2.1 の使用</b>	<b>171</b>
システム要件	172
プラットフォーム	172
監視できる eDirectory のバージョン	173
iMonitor へアクセスする	173
iMonitor のアーキテクチャ	173
iMonitor ページの構成	174
動作モード	175
すべてのページからアクセス可能な iMonitor の機能	176
NetWare Remote Manager との統合	176
環境設定ファイル	177
iMonitor の機能	179
eDirectory サーバのヘルス情報の表示	180
パーティション同期ステータスの表示	181
サーバ接続情報の表示	181
認識されているサーバの表示	182
レプリカ情報の表示	182
DS エージェントを制御および環境設定する	183
トレースを環境設定する	184
プロセスステータス情報の表示	185
エージェントアクティビティの表示	185
トラフィックパターンの表示	186
バックグラウンドプロセスの表示	186
eDirectory サーバエラーの表示	186
DSRepair 情報の表示	187
エージェントのヘルス情報の表示	187
ツリー内のオブジェクトの参照	187
同期またはパージのためのエントリの表示	188
DirXML の詳細の表示	188
レプリカの同期ステータスの表示	189
レポートの設定と表示	189
スキーマ、クラス、および属性定義の表示	190
オブジェクトの検索	191
ストリームビューアの使用	191
DIB セットのクローン	192
セキュリティ保護された iMonitor 操作の実現	196
<b>8 Novell eDirectory ツリーのマージ</b>	<b>197</b>
eDirectory ツリーのマージ	197
前提条件	198
ターゲットツリーの要件	198
スキーマの要件	198
ソースツリーをターゲットツリーへマージする	199
パーティションの変化	199
ソースツリーとターゲットツリーを準備する	200
マージする前の時刻の同期	201
2つのツリーをマージする	201
マージ後の作業	203
サーバツリーの結合	203



コンテキスト名の変更について . . . . .	206
ソースツリーとターゲットツリーを準備する . . . . .	206
ソースツリーとターゲットツリーを結合する . . . . .	208
ツリー名の変更 . . . . .	209
eMBox クライアントを使用したツリーのマージ . . . . .	210
DSMerge eMTool を使用する . . . . .	210
DSMerge eMTool オプション . . . . .	211
<b>9 Novell eDirectory データベースの修復 . . . . .</b>	<b>213</b>
基本修復操作の実行 . . . . .	214
無人修復を実行する . . . . .	215
ローカルデータベースの修復の実行 . . . . .	217
外部参照のチェック . . . . .	217
単一オブジェクトの修復 . . . . .	218
不明なリーフオブジェクトの削除 . . . . .	218
修復ログファイルの表示と設定 . . . . .	219
ログファイルを開く . . . . .	219
ログファイルオプションを設定する . . . . .	219
Novell iMonitor での修復の実行 . . . . .	219
レプリカの修復 . . . . .	220
すべてのレプリカを修復する . . . . .	220
選択したレプリカを修復する . . . . .	221
タイムスタンプを修復する . . . . .	221
このサーバを新しいマスタレプリカに設定する . . . . .	222
選択したレプリカを削除する . . . . .	223
レプリカリングを修復する . . . . .	223
すべてのレプリカリングを修復する . . . . .	223
選択したレプリカリングを修復する . . . . .	224
リング内のすべてのサーバにすべてのオブジェクトを送信する . . . . .	224
マスタから選択したレプリカへすべてのオブジェクトを受信する . . . . .	225
レプリカリングからこのサーバを削除する . . . . .	225
スキーマの保守 . . . . .	225
ツリーからスキーマを要求する . . . . .	226
ローカルスキーマをリセットする . . . . .	226
Post-NetWare 5 スキーマの更新を実行する . . . . .	227
オプションスキーマ拡張機能を実行する . . . . .	227
リモートスキーマをインポートする . . . . .	228
スキーマの新規エポックを宣言する . . . . .	228
サーバのネットワークアドレスの修復 . . . . .	229
すべてのネットワークアドレスを修復する . . . . .	229
サーバのネットワークアドレスを修復する . . . . .	229
同期化操作の実行 . . . . .	230
選択したレプリカをこのサーバで同期する . . . . .	230
このサーバの同期ステータスをレポートする . . . . .	231
すべてのサーバの同期ステータスをレポートする . . . . .	231
時刻同期を実行する . . . . .	231
即時同期をスケジュールリングする . . . . .	232
DSRepair のカスタムオプション . . . . .	232
eDirectory サーバ上で DSRepair を実行する . . . . .	233
DSRepair コマンドラインオプション . . . . .	234
DSRepair 詳細設定スイッチの使用 . . . . .	235
eMBox クライアントを使用したデータベースの修復 . . . . .	236
DSRepair eMTool を使用する . . . . .	236
DSRepair eMTool のオプション . . . . .	237

<b>10 WAN トラフィックマネージャ</b>	<b>241</b>
WAN トラフィックマネージャについて	241
LAN エリアオブジェクト	243
WAN トラフィックポリシー	244
WAN トラフィックを制限する	248
コストファクタを割り当てる	249
WAN トラフィックマネージャポリシーグループ	250
1-3AM.WMG	250
7AM-6PM.WMG	251
COSTLT20.WMG	251
IPX.WMG	251
NDSTTYP.S.WMG	252
ONOSPOOF.WMG	263
OPNSPOOF.WMG	263
SAMEAREA.WMG	264
TCPIP.WMG	264
TIMECOST.WMG	264
WAN ポリシーの構成	265
宣言セクション	265
セレクタセクション	267
プロバイダセクション	268
ポリシーセクションで使用される構文	268
<b>11 LDAP Services for Novell eDirectory について</b>	<b>273</b>
LDAP サービスの主な用語	274
クライアントとサーバ	274
オブジェクト	274
参照	275
LDAP と eDirectory の連携について	276
LDAP から eDirectory に接続する	277
クラスと属性のマッピング	280
非標準スキーマ出力を有効にする	283
構文の相違	284
サポートされる Novell LDAP コントロールおよび拡張	285
Linux、Solaris、AIX、または HP-UNIX 環境での LDAP ツールの使用	285
LDAP ツール	286
拡張可能一致検索フィルタ	295
<b>12 LDAP Services for Novell eDirectory の環境設定</b>	<b>299</b>
LDAP Services for eDirectory をロードおよびアンロードする	299
LDAP サーバがロードされているか確認する	300
LDAP サーバが実行されているか確認する	301
シナリオ	301
LDAP サーバが実行されているか確認する	302
デバイスがリスンしていることを確認する	304
LDAP オブジェクトを環境設定する	304
Linux、Solaris、AIX、HP-UX システム上で、LDAP サーバオブジェクトおよび LDAP グループオブジェクトを環境設定する	306
LDAP サーバをリフレッシュする	309
認証とセキュリティ	310
パスワードとの単純バインドに TLS を要求する	310
TLS を開始 / 停止する	311
TLS のサーバを環境設定する	312
TLS のクライアントを環境設定する	313
ルート認証局をエクスポートする	313
クライアント証明書で認証を受ける	314

サードパーティプロバイダの証明書を使用する	314
LDAP プロキシユーザを作成および使用する	314
SASL を使用する	316
LDAP サーバを使ってディレクトリを検索する	317
検索制限を設定する	317
参照を使用する	318
フィルタ済みレプリカを検索する	324
上方参照を設定する	324
シナリオ：連結ツリーでの上方参照	325
非委任領域を作成する	326
参照データを指定する	327
LDAP で参照情報を更新する	328
影響を受ける操作	328
上方参照のサポートの有無を確認する	328
持続的検索 :eDirectory イベントの設定	329
持続的検索の管理	329
イベントの監視拡張操作の使用を制御する	330
LDAP サーバの情報を取得する	331
<b>13 Novell eDirectory の SNMP サポート</b>	<b>335</b>
SNMP に関する用語の定義	335
SNMP サービスについて	336
eDirectory と SNMP	338
eDirectory の管理に SNMP を使う利点	338
eDirectory での SNMP の機能について	338
SNMP Services for eDirectory のインストールと設定	341
SNMP サーバモジュールのロードとアンロード	341
サブエージェントの設定	342
SNMP Services for eDirectory の設定	343
SNMP による eDirectory の監視	351
トラップ	351
トラップに関する設定	366
統計情報	375
トラブルシューティング	380
<b>14 Novell eDirectory のバックアップと復元</b>	<b>381</b>
eDirectory のバックアップ処理に関する確認事項	382
バックアップサービスおよび復元サービスについて	385
eDirectory Backup eMTool について	385
eDirectory 8.7.3 のバックアップ / 復元機能で変更された事項	386
Backup eMTool による復元作業の概要	388
バックアップファイルのヘッダ書式	389
バックアップログファイルの書式	393
DSMASTER サーバによる災害対策	394
遷移ベクトルと復元後の検証処理	395
復元後の検証については eDirectory 8.5 以降のみで互換性がある	395
NetWare のファイルシステムデータを復元する際のアクセス権の保存	396
ロールフォワードログを使用する	397
ロールフォワードログ機能を使用する上での注意事項	398
ロールフォワードログの保存先	399
ロールフォワードログのバックアップと削除	400
注意：eDirectory を削除するとロールフォワードログも削除される問題	401
復元処理の準備	401
復元作業の前提条件	402
復元に必要なバックアップファイルの収集	403
Novell iManager を使ったバックアップ / 復元作業	404

iManager による手動バックアップ	405
iManager によるロールフォワードログの設定	407
iManager によるバックアップファイルの復元作業	409
eMBox クライアントを使ったバックアップ/復元作業	412
eMBox クライアントによる手動バックアップ	412
バッチファイルと eMBox クライアントによる無人バックアップ	415
eMBox クライアントによるロールフォワードログの設定	418
eMBox クライアントによるバックアップファイルの復元作業	420
バックアップ/復元のコマンドラインオプション	423
サーバ固有情報のバックアップに関する変更事項 (NetWare のみ)	431
復元後の検証処理に失敗した場合の対処方法	432
レプリカリングをクリーンアップする	433
サーバの復旧とレプリカの再追加	435
バックアップ/復元の運用例	437
シナリオ: 単一サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合	437
シナリオ: 複数サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合	438
シナリオ: 複数サーバ構成のネットワークで、1 台のサーバが完全に使えなくなった場合	440
シナリオ: 複数サーバ構成のネットワークで、数台のサーバが使えなくなった場合	441
シナリオ: 複数サーバ構成のネットワークで、すべてのサーバが使えなくなった場合	441
<b>15 Novell eDirectory のメンテナンス</b>	<b>443</b>
eDirectory のパフォーマンスの改善	443
エントリキャッシュとブロックキャッシュでメモリを配分する	444
デフォルトのキャッシュ設定を使用する	444
LDAP for eDirectory をチューニングする	449
Linux、Solaris、AIX、および HP-UX システムでの eDirectory パフォーマンスの改善	451
eDirectory サーバを微調整する	451
eDirectory のキャッシュを最適化する	452
Novell eDirectory 用に Solaris OS をチューニングする	456
eDirectory の正常な動作の維持	457
ヘルスチェックを実行する時期	457
ヘルスチェックの概要	458
iMonitor を使用した eDirectory のヘルスチェック	458
詳細情報	460
監視のためのリソース	460
ハードウェアのアップグレードやサーバの交換	460
サーバを交換しないでハードウェアまたは記憶デバイスを計画的にアップグレードする	461
サーバの計画的な交換	464
ハードウェア障害後の eDirectory の復元	467
<b>16 DHost iConsole Manager</b>	<b>469</b>
DHost について	470
DHost iConsole の実行	471
NetWare で DHost iConsole を実行する	471
Windows で DHost iConsole を実行する	471
Linux、Solaris、AIX、および HP-UX で DHost iConsole を実行する	472
eDirectory モジュールの管理	472
NetWare でモジュールをロードまたはアンロードする	473
Windows でモジュールをロードまたはアンロードする	473
Linux、Solaris、AIX、および HP-UX でモジュールをロードまたはアンロードする	473
DHost 情報の照会	474
環境設定パラメータを表示する	474
プロトコル情報を表示する	475
接続プロパティを表示する	475
スレッドプールの統計情報を表示する	475
プロセススタック	476

SAdmin パスワードを設定する . . . . .	476
NetWare で SAdmin パスワードを設定する . . . . .	476
Windows で SAdmin パスワードを設定する . . . . .	477
Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する . . . . .	477
<b>17 eDirectory Management Toolbox . . . . .</b>	<b>479</b>
eMBox コマンドラインクライアントの使用 . . . . .	479
コマンドラインヘルプを表示する . . . . .	480
eMBox コマンドラインクライアントを対話式モードで実行する . . . . .	480
eMBox コマンドラインクライアントをバッチモードで実行する . . . . .	484
eMBox コマンドラインクライアントのオプション . . . . .	486
eMBox クライアントを使用してセキュア接続を確立する . . . . .	487
eDirectory ポート番号を確認する . . . . .	488
eMBox ログの記録の使用 . . . . .	489
eMBox ログの記録コマンドラインクライアントを使用する . . . . .	490
Novell iManager で eMBox ログの記録機能を使用する . . . . .	490
<b>18 Novell eDirectory のトラブルシューティング . . . . .</b>	<b>491</b>
エラーコードの解決 . . . . .	491
オブジェクトの命名 . . . . .	491
eDirectory バージョン番号の確認 . . . . .	492
NetWare の場合 . . . . .	492
Windows の場合 . . . . .	493
Linux の場合 . . . . .	493
Solaris の場合 . . . . .	494
AIX の場合 . . . . .	495
HP-UX の場合 . . . . .	495
eDirectory レプリカの問題から回復する . . . . .	496
eDirectory for Windows サーバが起動しない場合 . . . . .	496
Windows サーバが eDirectory データベースファイルを開けない場合 . . . . .	497
緊急修復後の Windows での eDirectory の復元 . . . . .	497
ログファイル . . . . .	498
modschema.log . . . . .	498
dsinstall.log . . . . .	498
LDIF ファイルのトラブルシューティング . . . . .	498
LDIF について . . . . .	498
LDIF ファイルのデバッグ . . . . .	506
LDIF を使用してスキーマを拡張する . . . . .	512
Linux、Solaris、AIX、および HP-UX での eDirectory のトラブルシューティング . . . . .	516
eDirectory への繰り返しログイン . . . . .	516
Novell パブリックキーインフラストラクチャサービス . . . . .	516
UNIX 上の NMAS . . . . .	517
LDAP サービス . . . . .	517
Novell インポート / エクスポート変換ユーティリティ . . . . .	518
ndsmerge ユーティリティ . . . . .	519
ndstrace ユーティリティ . . . . .	519
ndsbackup ユーティリティ . . . . .	519
インストールと設定 . . . . .	519
ndsrepair を使用する . . . . .	521
ndstrace の使用 . . . . .	528
ConsoleOne のトラブルシューティング . . . . .	536
eDirectory ツリーをブラウズできない . . . . .	536
NetWare サーバ上での純粋な IP 環境の実行 . . . . .	536
破損通知 . . . . .	537
例 . . . . .	538
停止および孤立した破損通知の影響 . . . . .	539

予防策	539
トラブルシューティングのヒント	540
DS がロードされない場合の HTTPSTK へのアクセス	543
NetWare で SAdmin パスワードを設定する	543
Windows で SAdmin パスワードを設定する	544
Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する	544
SNMP のトラブルシューティング	545
必要なトラップが生成されない	545
Linux 上の SNMP	545
HP-UX 上の SNMP	546
コンテナ管理者の eDirectory インストールの失敗	547
Sun One スキーマの Novell eDirectory への移行	547
手順 1: スキーマキャッシュの更新操作を実行する	547
手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する	548
手順 3: LDIF ファイルのインポート	550
ICE を使用した、アクティブディレクトリスキーマの Novell eDirectory への移行	550
手順 1: スキーマキャッシュの更新操作を実行する	551
手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する	551
手順 3: LDIF ファイルのインポート	552
<b>A NMAS の注意事項</b>	<b>553</b>
独立したパーティションとしてのセキュリティコンテナの設定	553
複数のセキュリティコンテナを持つツリーのマージ	553
ツリーのマージ前に実行する製品固有の操作	554
ツリーのマージを実行する	557
ツリーのマージ後に実行する製品固有の操作	557
<b>B Novell eDirectory の UNIX コマンドとその使用法</b>	<b>559</b>
一般コマンド	559
LDAP 固有のコマンド	561
<b>C OpenSLP for eDirectory の設定</b>	<b>563</b>
Service Location Protocol	563
SLP の基本	563
Novell Service Location Providers	564
ユーザエージェント	565
サービスエージェント	565
環境設定パラメータ	566
<b>D Novell eDirectory が DNS を使用する際の動作について</b>	<b>567</b>

## このガイドについて

このガイドでは、Novell® eDirectory™ 8.7.3 の管理および設定の方法について説明します。このガイドはネットワーク管理者を対象にし、次のセクションから構成されています。

- ◆ 1 章 13 ページの、「Novell eDirectory について」
- ◆ 2 章 67 ページの、「Novell eDirectory ネットワークの設計」
- ◆ 3 章 89 ページの、「オブジェクトの管理」
- ◆ 4 章 105 ページの、「スキーマの管理」
- ◆ 5 章 117 ページの、「パーティションおよびレプリカの管理」
- ◆ 6 章 131 ページの、「Novell eDirectory 管理ユーティリティ」
- ◆ 7 章 171 ページの、「Novell iMonitor 2.1 の使用」
- ◆ 8 章 197 ページの、「Novell eDirectory ツリーのマージ」
- ◆ 9 章 213 ページの、「Novell eDirectory データベースの修復」
- ◆ 10 章 241 ページの、「WAN トラフィックマネージャ」
- ◆ 11 章 273 ページの、「LDAP Services for Novell eDirectory について」
- ◆ 12 章 299 ページの、「LDAP Services for Novell eDirectory の環境設定」
- ◆ 13 章 335 ページの、「Novell eDirectory の SNMP サポート」
- ◆ 14 章 381 ページの、「Novell eDirectory のバックアップと復元」
- ◆ 15 章 443 ページの、「Novell eDirectory のメンテナンス」
- ◆ 16 章 469 ページの、「DHost iConsole Manager」
- ◆ 17 章 479 ページの、「eDirectory Management Toolbox」
- ◆ 18 章 491 ページの、「Novell eDirectory のトラブルシューティング」
- ◆ 553 ページの付録 A、「NMAP の注意事項」
- ◆ 559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」
- ◆ 563 ページの付録 C、「OpenSLP for eDirectory の設定」
- ◆ 567 ページの付録 D、「Novell eDirectory が DNS を使用する際の動作について」

### 補足マニュアル

eDirectory のインストール手順については、『*Novell eDirectory 8.7.3 インストールガイド* (<http://www.novell.com/documentation/japanese/edir873/index.html>)』を参照してください。

eDirectory 管理ユーティリティに関するマニュアルについては、『*Novell iManager 2.0.x Administration Guide(Novell iManager 2.0.x 管理ガイド)* (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

## マニュアルの更新

このガイドの最新版については、『*Novell eDirectory 8.7.3 管理ガイド* (<http://www.novell.com/documentation/japanese/edir873/index.html>)』を参照してください。

## マニュアルの表記規則

このマニュアルでは、不等号 (>) を使用して、操作手順の動作、およびクロスリファレンスパス内の項目を区切ります。

「®」、「™」などの商標記号は、Novell の商標を示します。アスタリスク (\*) はサードパーティの商標を示します。

シングルパス名に円記号 (¥) が使用されるプラットフォームやスラッシュ (/) が使用されるプラットフォームがありますが、パス名は円記号で表記されています。UNIX\* など、スラッシュを必要とするプラットフォームでは、ソフトウェアの必要に応じてスラッシュを使用してください。



# 1

## Novell eDirectory について

Novell® eDirectory™ は、高い拡張性を持ち、高性能で、安全性の高いディレクトリサービスです。ユーザ、アプリケーション、ネットワークデバイス、およびデータなど、多量のオブジェクトを格納および管理できます。Novell eDirectory が提供する安全な識別情報管理ソリューションは、複数のプラットフォーム間で実行され、インターネット規模の高い拡張性を持っています。

また、識別情報管理、インフラストラクチャ、インターネット上のセキュリティ、および拡張性が 1 つに集約されて、ファイアウォール越しに実行されるすべてのアプリケーションで利用できます。Novell eDirectory 8.7.3 には、Web ベースのワイヤレスな管理機能が搭載されており、Web ブラウザおよびさまざまな携帯用デバイスから、ディレクトリやユーザ、アクセス権、およびネットワークリソースにアクセスしたり、それら进行管理したりすることができます。

eDirectory では、ディレクトリ標準 LDAP(Lightweight Directory Access Protocol) バージョン 3 がネイティブでサポートされ、OpenSSL ソースコードに基づいた TLS/SSL サービスへのサポートが提供されています。

eDirectory エンジンの詳細については、[eDirectory Process Requests \(http://developer.novell.com/research/sections/netmanage/dirprimer/2002/august/p020801.htm\)](http://developer.novell.com/research/sections/netmanage/dirprimer/2002/august/p020801.htm) を参照してください。

このセクションでは、次の情報について説明します。

- ◆ 14 ページの「新機能」
- ◆ 14 ページの「Novell eDirectory」
- ◆ 15 ページの「Novell iManager の便利な管理機能」
- ◆ 19 ページの「オブジェクトクラスとプロパティ」
- ◆ 35 ページの「コンテキストと命名規則」
- ◆ 38 ページの「スキーマ」
- ◆ 44 ページの「パーティション」
- ◆ 48 ページの「レプリカ」
- ◆ 52 ページの「NetWare バインダリエミュレーション」
- ◆ 53 ページの「レプリカリングでのサーバの同期」
- ◆ 53 ページの「リソースへのアクセス」
- ◆ 54 ページの「eDirectory での権利」

## 新機能

Novell eDirectory 8.7.3 には、次のような新機能があります。

- ◆ 次のプラットフォームへの新規サポート
  - ◆ Windows\* 2003 Server
- ◆ Linux\*、Solaris\*、および AIX\* のすべての eDirectory サーバコンポーネントに対する、UNIX\* パッケージベースのインストール

詳細については、『*Novell eDirectory 8.7.3 インストールガイド*』を参照してください。

- ◆ デフォルトポートを 80 および 443 から、8008 および 8010 (NetWare<sup>®</sup> では 8009) に変更  
詳細については、[488 ページの「eDirectory ポート番号を確認する」](#)を参照してください。

- ◆ Novell iManager 2.0.2

NetWare 6.5、Windows、Linux、Solaris、AIX、および HP-UX\* 上で Novell 製品の管理を行うための、単一の Web ベース管理コンソールを搭載しています。iManager では、Novell の Web ベース管理ユーティリティが、すべて 1 つの管理フレームワーク上で標準化されています。また、オープンな標準アプリケーションインタフェースによって、Web ベース管理運用モジュールの開発を容易にするための最高のアーキテクチャも用意されています。

詳細については、『*Novell iManager 2.0.x Administration Guide (Novell iManager 2.0.x 管理ガイド)* (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

- ◆ NMAS™ 2.3

Novell Modular Authentication Service™ (NMAS) の新機能には、高度なパスワードポリシーの実行、NMAS Web サーバエージェント、チャレンジ/レスポンス方式のログインメソッド、チャレンジレスポンス認証 API、Kerberos 認証メソッドなどがあります。

詳細については、『*Novell Modular Authentication Service 2.3 Administration Guide (Novell Modular Authentication Service 2.3 管理ガイド)* (<http://www.novell.com/documentation/japanese/nmas23/index.html>)』を参照してください。

- ◆ Novell Certificate Server 2.6

新機能には、SSL 上の OCSP およびディレクトリ名 CRL サポートなどがあります。

詳細については、『*Novell Certificate Server 2.6 Administration Guide (Novell Certificate Server 2.6 管理ガイド)* (<http://www.novell.com/documentation/japanese/crt26/index.html>)』を参照してください。

## Novell eDirectory

簡単に言うと、Novell eDirectory とは、ネットワークユーザ、サーバ、プリンタ、プリントキュー、アプリケーションなどのネットワークリソースを表す一連のオブジェクトです。[図 1](#) は、Novell iManager 管理ユーティリティで表示されるオブジェクトの一部を示します。

図 1 iManager での eDirectory オブジェクト



eDirectory サーバに設定された実際のスキーマや eDirectory を実行するオペレーティングシステムによっては、使用できないオブジェクトクラスもあります。

オブジェクトの詳細については、19 ページの「オブジェクトクラスとプロパティ」を参照してください。

ディレクトリは、サーバのデータベースファイルのセットとして、物理的に格納されません。サーバがファイルシステムボリュームのホストとなっている場合、ファイルはボリューム `sys:` に格納されます。ボリュームがまったくない場合、ディレクトリはサーバのローカルディスクに格納されます。

ネットワークに複数の eDirectory サーバがある場合、ディレクトリは複数のサーバに複製できます。

## Novell iManager の便利な管理機能

Novell eDirectory では、ネットワークリソースを、容易に、強力に、またフレキシブルに管理できます。eDirectory は、グループウェアやその他アプリケーションのユーザ情報のリポジトリとしても機能します。これらのアプリケーションは、業界標準の LDAP(Lightweight Directory Access Protocol) を通して、ディレクトリへアクセスします。

eDirectory の便利な管理機能には、強力なツリー構造、統合管理ユーティリティ、およびシングルログインと認証機能があります。

Novell iManager では、Web ブラウザやさまざまな携帯用デバイスから、ディレクトリやユーザ、ディレクトリ内のアクセス権やネットワークリソースを管理できます。iManager への eDirectory プラグインによって、基本的なディレクトリ管理タスクや、DSRepair、DSMerge、バックアップおよび復元などの、以前は eDirectory サーバで実行する必要のあった eDirectory 管理ユーティリティにアクセスすることができます。

iManager が Web サーバにインストールされると、Internet Explorer 5.5 以降または Netscape 6.2 以降を実行するサーバやワークステーションであればどこからでもアクセスできます。

詳細については、『[Novell iManager 2.0.x Administration Guide\(Novell iManager 2.0.x 管理ガイド\)](http://www.novell.com/documentation/japanese/imanager20/index.html)』( <http://www.novell.com/documentation/japanese/imanager20/index.html> ) を参照してください。

## 強力なツリー構造

Novell eDirectory では、ツリー構造内にオブジェクトが編成されます。最上部のツリーオブジェクトには、ツリー名が付けられます。

eDirectory サーバで動作するオペレーティングシステムが NetWare、UNIX、Windows のどれであっても、すべてのリソースを同じツリー内に保管できます。オブジェクトの作成や、権利の許可、パスワードの変更、アプリケーションの管理のために、それぞれのサーバやドメインに個別にアクセスする必要はありません。

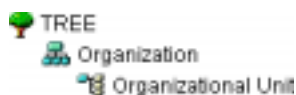
ツリーの階層構造によって、極めて柔軟で強力な管理が可能になっています。このような管理機能は、主に次の 2 つの機能で実現できます。


- ◆ 16 ページの「コンテナオブジェクト」
- ◆ 17 ページの「継承」


## コンテナオブジェクト


コンテナオブジェクトを利用することによって、オブジェクトを個々に扱うのではなく、オブジェクトのまとまりとして扱うことができます。図 2 に示すように、コンテナオブジェクトには、3 つの共通クラスがあります。

図 2 コンテナオブジェクトの共通クラス




 ツリーオブジェクトは、ツリー内の最上部のコンテナオブジェクトです。通常、このコンテナには、その会社の組織オブジェクトが格納されます。

 組織は通常、ツリーオブジェクトの直下に位置するコンテナクラスです。一般に、組織オブジェクトには会社名に基づく名前が付けられます。小規模な会社の場合は、管理を容易化するために、他のオブジェクトをすべて組織オブジェクトの直下に配置します。

 組織の下に部門オブジェクトを作成し、異なった地区、ネットワークキャンパス、または個々の部署を表すことができます。部門の下にさらに部門を作成して、ツリーを細分化することもできます。

その他、コンテナオブジェクトには、国クラスと地域クラスがあります。これらのクラスは通常、複数の国にまたがるネットワークでのみ使用されます。

 ドメインオブジェクトは、ツリーオブジェクトの下、または組織、部門、国、および地域オブジェクトの下に作成されます。

コンテナ内のすべてのオブジェクトに対してなんらかの処理を行う場合、コンテナオブジェクトに処理を行えば 1 回の実行で済みます。たとえば、Accounting コンテナ内のすべてのオブジェクトに対する管理制御権を、Amy という名前のユーザに与えるとしします。(図 3 を参照してください)。

図 3 コンテナオブジェクト



この場合、Accounting オブジェクトを右クリックして [オブジェクトのトラスティ] を選択し、Amy をトラスティとして追加します。次に、Amy に与える権利を選択して [OK] をクリックします。これで Amy に、Database アプリケーション、Bookkeepers グループ、LaserPrinter プリンタ、および Amy、Bill、Bob のユーザオブジェクトを管理する権利が与えられました。

## 継承

eDirectory のもう 1 つの強力な機能は、権利の継承です。継承とは、ツリー内の上位層のコンテナの権利が、それぞれの下位層のすべてのコンテナに受け継がれることを意味します。この機能によって、権利の割り当て回数を少なくすることができます。たとえば、図 4 に示すオブジェクトに対する管理権を与えるとします。

図 4 eDirectory オブジェクトの例



次の割り当てのいずれかを行うことができます。

- ◆ ユーザに Allentown に対する権利を与えると、そのユーザは Allentown コンテナ内のオブジェクトのみ管理できます。
- ◆ ユーザに East に対する権利を与えると、そのユーザは East、Allentown、および Yorktown コンテナ内のオブジェクトを管理できます。
- ◆ ユーザに YourCo に対する権利を与えると、そのユーザは図中のすべてのコンテナのすべてのオブジェクトを管理できます。

権利の割り当ての詳細については、54 ページの「eDirectory での権利」を参照してください。

## Web ベースの管理ユーティリティ

Novell iManager はブラウザベースのツールで、eDirectory オブジェクトを運用、管理、設定するために使用します。iManager を使用すると、ユーザに特定のタスクや責任を割り当てたり、それらのタスクを実行するために必要なツールおよびそれに伴う権利だけを付与したりすることができます。

Internet Explorer 5.5 SP2 以降または Netscape 6.2 以降を実行するサーバまたはワークステーションで iManager を使用すると、次のようなスーパーバイザの作業を実行できます。

- ◆ eDirectory への LDAP および XML ベースのアクセス設定
- ◆ ネットワークユーザ、デバイス、およびリソースを表すオブジェクトの作成
- ◆ 新規ユーザアカウント作成用のテンプレートの定義
- ◆ ネットワークオブジェクトの検索、変更、移動、および削除
- ◆ 管理権を委託する権利と職種の定義
- ◆ カスタムオブジェクトタイプとプロパティを作成するための eDirectory スキーマの拡張
- ◆ 複数のサーバでの eDirectory データベースのパーティション化とレプリカの作成
- ◆ DSRepair、DSMerge、バックアップおよび復元などの eDirectory 管理ユーティリティの実行

その他にも、iManager にロードされたプラグインに基づいた管理機能を実行できます。iManager 1.5 でインストールされるのは、次の eDirectory プラグインです。

- ◆ eDirectory Backup and Restore
- ◆ eDirectory Log Files
- ◆ eDirectory Merge
- ◆ eDirectory Repair
- ◆ eDirectory Service Manager
- ◆ eGuide コンテンツ
- ◆ iManager 基本コンテンツ
- ◆ インポート / エクスポート変換ウィザード
- ◆ インデックス管理
- ◆ iPrint
- ◆ LDAP
- ◆ NLS
- ◆ NMAS
- ◆ PKI/Certificate
- ◆ フィルタ処理済レプリカ環境設定ウィザード
- ◆ SNMP
- ◆ WAN トラフィックマネージャ

iManager のインストール、設定、および実行についての詳細は、『[Novell iManager 2.0.x 管理ガイド](http://www.novell.com/documentation/japanese/imanager20/index.html) (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

## シングルログインと認証

eDirectory では、ユーザはグローバルディレクトリにログインするため、ユーザごとに複数のサーバやドメインを管理する必要はありません。また、ドメイン間の信頼関係や通過時の認証を管理する必要もありません。

ディレクトリのセキュリティ機能の 1 つは、ユーザの認証です。ユーザがログインするためには、あらかじめユーザオブジェクトがディレクトリ内に作成されている必要があります。ユーザオブジェクトには、名前やパスワードといった、特定のプロパティがあります。

ユーザがログインすると、eDirectory は入力されたパスワードとディレクトリに格納されているそのユーザのパスワードを照合し、一致した場合にアクセスを許可します。

## オブジェクトクラスとプロパティ

eDirectory オブジェクトの各タイプの定義を、オブジェクトクラスといいます。たとえば、「ユーザ」や「組織」は、オブジェクトクラスです。オブジェクトの各クラスには、それぞれ特定のプロパティがあります。たとえば、ユーザオブジェクトには、ログイン名、パスワード、姓、および他の多くのプロパティがあります。



スキーマでは、オブジェクトクラスとプロパティ、および保管規則 ( どのコンテナにどのオブジェクトを保管するか ) が定義されます。eDirectory にはベーススキーマが付属しています。このベーススキーマは、ユーザまたはユーザが使用するアプリケーションによる拡張が可能です。スキーマの詳細については、[38 ページの「スキーマ」](#)を参照してください。





コンテナオブジェクトは、他のオブジェクトを格納し、ツリーをさまざまな分岐に分割するために使用されます。一方、リーフオブジェクトはネットワークリソースを表します。

## オブジェクトのリスト





次の表に、eDirectory のオブジェクトクラスを示します。サービスを追加した場合は、表内のオブジェクトクラス以外のオブジェクトクラスが eDirectory 内に新たに作成されることがあります。また、eDirectory のホストとなるすべてのサーバオペレーティングシステムで、すべてのクラスが使用できるわけではありません。

### eDirectory コンテナオブジェクトクラス

iManager アイコン	コンテナオブジェクト (略語)	説明
	ツリー	ツリーの開始点を表します。詳細については、 <a href="#">22 ページの「ツリー」</a> を参照してください。
	国 (C)	ネットワークが存在する国を表します。その下には、国内の他のディレクトリオブジェクトが編成されます。詳細については、 <a href="#">24 ページの「国」</a> を参照してください。

iManager アイコン	コンテナオブジェクト (略語)	説明
	ライセンスコンテナ (LC)	NLS(Novell Licensing Services) 技術を使用して、ライセンス許可証をインストールした場合や課金許可証を作成した場合に自動的に作成されます。NLS 対応のアプリケーションをインストールすると、LC コンテナオブジェクトがツリーに追加され、ライセンス許可証リーフオブジェクトがそのコンテナに追加されます。
	組織 (O)	ディレクトリ内の他のオブジェクトの編成に使用されます。組織オブジェクトは、国オブジェクトの直下に配置されます ( 国オブジェクトを作成している場合 )。詳細については、 <a href="#">22 ページの「組織」</a> を参照してください。
	部門 (OU)	ディレクトリ内の他のオブジェクトをさらに細かく編成するために使用されます。部門オブジェクトは、組織オブジェクトの直下に配置されます。詳細については、 <a href="#">23 ページの「部門」</a> を参照してください。
	ドメイン (DC)	ディレクトリ内の他のオブジェクトをさらに細かく編成するために使用されます。ドメインオブジェクトは、ツリーオブジェクトの下、または組織、部門、国、および地域オブジェクトの下に作成されます。詳細については、 <a href="#">24 ページの「ドメイン」</a> を参照してください。

## eDirectory リーフオブジェクトクラス

iManager アイコン	リーフオブジェクト	説明
	AFP サーバ	eDirectory ネットワーク内のノードとして機能する、AppleTalk* ファイリングプロトコルサーバを表します。通常、複数の Macintosh* コンピュータに対する NetWare ルータおよび AppleTalk サーバとしても機能します。
	別名	ディレクトリ内にあるオブジェクトの実際の位置を指します。別名を使用することによって、ディレクトリ内のディレクトリオブジェクトを、実際の場所とは異なる場所に存在するように表示できます。詳細については、 <a href="#">32 ページの「別名」</a> を参照してください。
	アプリケーション	ネットワークアプリケーションを表します。アプリケーションオブジェクトによって、権限の割り当て、ログインスクリプトのカスタマイズ、およびアプリケーションの起動のような、管理作業を簡素化できます。
	コンピュータ	ネットワーク内のコンピュータを表します。



iManager アイコン	リーフオブジェクト	説明
	ディレクトリマップ	ファイルシステム内のディレクトリを表します。詳細については、 <a href="#">34 ページの「ディレクトリマップ」</a> を参照してください。
	グループ	ディレクトリ内のユーザオブジェクトのリストに名前を割り当てます。各ユーザに権利を割り当てる代わりに、グループに権利を割り当てることによって、グループ内の各ユーザに権利を与えることができます。詳細については、 <a href="#">29 ページの「グループ」</a> を参照してください。
	ライセンス許可証	プロダクトライセンス許可証をデータベースのオブジェクトとしてインストールするために、NLS 技術とともに使用されます。NLS 対応アプリケーションをインストールすると、ライセンス許可証オブジェクトがライセンスプロダクトコンテナに追加されます。
	職種	組織内での地位や職種を定義します。
	プリントキュー	ネットワークのプリントキューを表します。
	プリントサーバ	ネットワークのプリントサーバを表します。
	プリンタ	ネットワークのプリンタを表します。
	プロファイル	共通のログインスクリプトコマンドを共有するユーザグループが使用するログインスクリプトを表します。これらのユーザは同じコンテナに属する必要はありません。詳細については、 <a href="#">35 ページの「プロファイル」</a> を参照してください。
	サーバ	任意のオペレーティングシステムが動作するサーバを表します。詳細については、 <a href="#">25 ページの「サーバ」</a> を参照してください。
	テンプレート	新しいユーザオブジェクトに適用する標準のユーザオブジェクトプロパティを表します。
	不明	iManager にカスタムアイコンが存在しないオブジェクトを表します。
	ユーザ	ネットワークを使用する人を表します。詳細については、 <a href="#">27 ページの「ユーザ」</a> を参照してください。
	ボリューム	ネットワーク上の物理的なボリュームを表します。詳細については、 <a href="#">26 ページの「ボリューム」</a> を参照してください。

# コンテナオブジェクトクラス

## ツリー



ネットワーク内のサーバに eDirectory を初めてインストールすると、ツリーコンテナ (以前の [Root] コンテナ) が作成されます。最上位のコンテナであるツリーコンテナには、通常、組織オブジェクト、国オブジェクト、または別名オブジェクトが格納されます。

### ツリーが表す内容

ツリーコンテナはツリーの最上部を表します。

### 使用法

ツリーは、包括的な権利の割り当てに使用します。ツリーに対して行った権利の割り当ては、継承機能によって、ツリー内のすべてのオブジェクトに適用されます。54 ページの「eDirectory での権利」を参照してください。デフォルトで、トラスティ [Public] はツリーに対するブラウズ権を所有し、Admin はツリーに対するスーパーバイザ権を所有します。

### 重要なプロパティ

ツリーオブジェクトは名前プロパティを持っています。名前プロパティは、最初のサーバのインストール時に指定されたツリー名を表します。ツリー名は iManager の階層に表示されます。

## 組織



ネットワークのサーバに eDirectory をインストールすると、組織コンテナオブジェクトが作成されます。通常、組織コンテナは最上部のツリーコンテナの直下に作成され、コンテナ内には部門オブジェクトとリーフオブジェクトが格納されます。

デフォルトでは、最初の組織コンテナに、Admin という名のユーザオブジェクトが作成されます。

### 組織オブジェクトが表す内容

通常、組織オブジェクトは会社を表しますが、ツリーの下に組織オブジェクトを追加作成することもできます。一般的に、組織オブジェクトの追加作成は、さまざまな地区で構成されるネットワークや、独立した複数の eDirectory ツリーがマージされているネットワークで行われます。

### 使用法

ツリーでの組織オブジェクトの運用方法は、ネットワークのサイズと構造により異なります。小規模のネットワークでは、1 つの組織オブジェクトの下にすべてのリーフオブジェクトを配置します。

大規模なネットワークでは、組織オブジェクトの下に部門オブジェクトを作成します。これにより、リソースの検索と管理を容易にできます。たとえば、社内の各部署や事業部ごとに部門オブジェクトを作成できます。

複数のサイトがあるネットワークでは、組織オブジェクトの下に各サイトを表す部門オブジェクトを作成します。ディレクトリを分割するためのサーバ数が十分にあれば、このようにサイトの境界で論理的にパーティションを区切ることができます。

プリンタ、ボリューム、アプリケーションといった、社内全体で使用するリソースを共有しやすくするために、組織の直下に、対応するプリンタ、ボリューム、アプリケーションのオブジェクトを作成します。

## 重要なプロパティ

組織オブジェクトの最も有用なプロパティを次に示します。名前プロパティは必須です。すべてのプロパティの一覧を表示するには、iManager で組織オブジェクトを選択します。プロパティの各ページの説明を表示するには、[ヘルプ] をクリックします。

### ◆ 名前


通常、名前プロパティは会社名と同じです。もちろん、簡略化するために短くすることもできます。たとえば、会社名が **Your Shoe Company** の場合、**YourCo** とすることができま

す。組織名は、その下に作成されるすべてのオブジェクトのコンテキストの一部として使用されます。

### ◆ ログインスクリプト

ログインスクリプトプロパティには、組織の直下にあるユーザオブジェクトが実行するコマンドが格納されます。これらのコマンドは、ユーザのログイン時に実行されます。

## 部門

 **部門 (OU)** コンテナオブジェクトを作成することによって、ツリーを細分化できます。部門は、iManager によって、組織、国、または別の部門オブジェクトの下に作成されます。

部門には、ユーザオブジェクトやアプリケーションオブジェクトといった、他の部門やリーフオブジェクトを格納できます。

## 部門オブジェクトが表す内容

通常は、部門オブジェクトは1つの部署を表し、互いにアクセスする必要がある複数のオブジェクトを格納します。部門オブジェクトの主な格納内容として、複数のユーザ、およびユーザが使用するプリンタ、ボリューム、アプリケーションなどを挙げる事ができます。

部門オブジェクトの最上位レベルに配置された各部門は、WAN リンクごとに区切られたネットワークの各サイトを表します。

## 使用法

ツリーでの部門オブジェクトの運用方法は、ネットワークのサイズと構造により異なります。小規模のネットワークでは、部門オブジェクトを作成する必要がない場合もあります。

大規模なネットワークでは、組織オブジェクトの下に部門オブジェクトを作成します。これにより、リソースの検索と管理を容易にできます。たとえば、社内の各部署や事業部ごとに部門オブジェクトを作成できます。ユーザオブジェクトと、ユーザが頻繁に使用するリソースを一緒に部門オブジェクトに格納すると、管理が最も容易になります。

複数のサイトがあるネットワークでは、組織オブジェクトの下に各サイトを表わす部門オブジェクトを作成します。ディレクトリを分割するためのサーバ数が十分にあれば、このようにサイトの境界で論理的にパーティションを区切ることができます。

## 重要なプロパティ

部門オブジェクトの最も有用なプロパティを次に示します。名前プロパティは必須です。すべてのプロパティの一覧を表示するには、iManager で部門オブジェクトを選択します。プロパティの各ページの説明を表示するには、[ヘルプ] をクリックします。

- ◆ 名前


通常、名前プロパティは部署名と同じです。もちろん、簡略化するために短くすることもできます。たとえば、部署名が **Accounts Payable** の場合、省略して **AP** とすることができます。

部門名は、その下に作成されるすべてのオブジェクトのコンテキストの一部として使用されます。

- ◆ ログインスクリプト

ログインスクリプトプロパティには、部門の直下にあるユーザオブジェクトが実行するコマンドが格納されます。これらのコマンドは、ユーザのログイン時に実行されます。

## 国

 国オブジェクトは、iManager を使用して、ツリーオブジェクトの直下に作成できます。国オブジェクトは、特定の X.500 グローバルディレクトリに接続する場合にのみ必要です。

### 国オブジェクトが表す内容

国オブジェクトは、ツリーの分岐先の国家名を表します。

### 使用法


ネットワークが複数の国に渡っている場合でも、管理者は通常、国オブジェクトを作成しません。これは、国オブジェクトがツリーに不要なレベルを追加するだけだからです。ネットワークが複数の国家で構成されている場合は、必要に応じて、ツリーオブジェクトの下に 1 つ以上の国オブジェクトを作成できます。国オブジェクトには、組織オブジェクトのみ格納できます。

国オブジェクトを作成していない場合でも、後で必要になった時には、随時ツリーを変更して国オブジェクトを追加できます。

## 重要なプロパティ

国オブジェクトには、2 文字の名前プロパティがあります。国オブジェクトの名前には、US、UK、または DE といった、2 文字の標準コードが使用されます。

## ドメイン

 ドメインオブジェクトは、iManager を使用してツリーオブジェクトの直下に作成できます。また、組織、部門、国、および地域オブジェクトの下にも作成できます。

### ドメインオブジェクトが表す内容

ドメインオブジェクトは、DNS のドメインコンポーネントを表します。ドメインオブジェクトを使用すると、ドメインネームシステムによって示されるサービスリソースレコードの場所 (DNS SRV) に基づいて、ツリー内のサービスを検索できます。

ドメインオブジェクトを使用すると、ツリーは次のように表されます。

DS=Novell.DC=Provo.DC=USA

この例では、すべてのサブコンテナがドメインになっています。次のように、異なるツリーが混在する場合にもドメインオブジェクトを使用できます。

DC=Novell.O=Provo.C=USA

または

OU=Novell.DC=Provo.C=USA

通常、先頭のドメインはツリー全体を表し、サブドメインはそのツリーの下位の部分を表します。たとえば `machine1.novell.com` をツリーで表すと、`DC=machine1.DC=novell.DC=com` となります。ドメインは、eDirectory ツリーの設定で使用される一般的な方法です。コンテナおよびサブコンテナがすべて DC オブジェクトである場合は、オブジェクトを検索するときに、C、O、または OU を記憶している必要はありません。


## 使用法

NetWare 4 および 5 のツリーでは、ドメインオブジェクトをツリーの最上位にすることはできません。NetWare 4 および 5 では、NCP サーバオブジェクトを組織、国、部門、または地域コンテナに配置できますが、ドメインコンテナには配置できません。ただし NetWare 6 では、ドメインオブジェクトをツリーの最上位にすることができます。また NCP サーバオブジェクトをドメインコンテナに配置できます。

NetWare の以前のインストール (NetWare 4 など) を使用している場合、NetWare 5 以降のインストールやアップグレードに備えてツリーを設定するときに、`nds500.sch` ファイルが自動的に実行されます。最初のサーバをツリーにインストールすると、このファイルによりスキーマが拡張され、任意の場所にドメインコンテナを作成し、ほとんどのディレクトリオブジェクトを格納できます。

## リーフオブジェクトクラス

### サーバ

 サーバに eDirectory をインストールすると、そのサーバに対応するサーバオブジェクトが、ツリー内に自動作成されます。このオブジェクトクラスは、eDirectory が動作しているいずれかのサーバを表します。

NetWare 2 または NetWare 3 バインダリサーバを表すサーバオブジェクトを作成することもできます。

### サーバオブジェクトが表す内容

サーバオブジェクトは、eDirectory が動作しているサーバ、またはバインダリベース (NetWare 2 または NetWare 3) のサーバを表します。

### 使用法


サーバオブジェクトはレプリケーション処理のリファレンスポイントの役目を果たします。バインダリベースのサーバを表すサーバオブジェクトでは、iManager でそのサーバのボリュームを管理できます。

## 重要なプロパティ

サーバオブジェクトの主なプロパティとして、ネットワークアドレスプロパティがあります。ネットワークアドレスプロパティには、そのサーバのプロトコルとアドレス番号が表示されます。これはパケットレベルでのトラブルシューティングに役立ちます。

すべてのプロパティの一覧を表示するには、iManager でサーバオブジェクトを選択します。プロパティの各ページの説明を表示するには、[ヘルプ] をクリックします。

## ボリューム

 サーバ上に物理ボリュームを作成すると、ツリー内にボリュームオブジェクトが自動作成されます。デフォルトでは、サーバ名にアンダースコアと物理ボリューム名を追加したものが、ボリュームオブジェクトの名前になります (YOSERVER\_SYS など)。

ボリュームオブジェクトは NetWare でのみ作成できます。UNIX ファイルシステムのパーティションは、ボリュームオブジェクトを使用して管理することはできません。

### ボリュームオブジェクトが表す内容

ボリュームオブジェクトは、サーバ上の物理ボリューム (書き込み可能ディスクや CD などの記憶媒体) を表します。eDirectory 内のボリュームオブジェクトには、そのボリューム内のファイルやディレクトリに関する情報は含まれませんが、iManager を使用すれば、それらの情報にアクセスできます。ファイルおよびディレクトリに関する情報は、ファイルシステム自体に保存されます。

### 使用法

iManager で [ボリューム] アイコンをクリックすると、そのボリュームのファイルやディレクトリを管理できます。iManager により、ボリュームの空きディスク容量、ディレクトリエントリ領域、および圧縮の統計に関する情報が提供されます。


ツリー内に、NetWare 2 や NetWare 3 ボリュームのボリュームオブジェクトを作成することもできます。

## 重要なプロパティ

必須の名前プロパティおよびホストボリュームプロパティに加えて、ボリュームオブジェクトには他にも重要なプロパティがあります。

- ◆ 名前  
ツリー内のボリュームオブジェクトの名前です。デフォルトでは、この名前は物理ボリュームの名前に基づいて付けられますが、変更も可能です。
- ◆ ホストサーバ  
ボリュームが存在するサーバの名前です。
- ◆ バージョン  
ボリュームの格納先であるサーバの NetWare または eDirectory のバージョンです。
- ◆ ホストボリューム  
物理ボリューム名です。実際のボリュームオブジェクト名は物理ボリューム名を反映する必要がないため、ボリュームオブジェクトを物理ボリュームと関連付けるためにこのプロパティが必要になります。

## ユーザ

 ログインにはユーザオブジェクトが必要です。ツリーに最初のサーバがインストールされると、Admin というユーザオブジェクトが作成されます。初回ログイン時には、Admin としてログインします。

ユーザオブジェクトの作成またはインポートには、次の機能を使用できます。

- ◆ iManager

iManager の詳細については、『*Novell iManager 2.0.x Administration Guide(Novell iManager 2.0.x 管理ガイド)* (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

- ◆ データベースファイルからのバッチ処理

バッチファイルの使用の詳細については、68 ページの「eDirectory ツリーの設計」を参照してください。

- ◆ NetWare アップグレードユーティリティ

既存のバインダリサーバからのユーザのインポートなど、アップグレードユーティリティの詳細については、68 ページの「eDirectory ツリーの設計」を参照してください。

### ユーザオブジェクトが表す内容

ユーザオブジェクトはネットワークを使用するユーザを表します。

### 使用法

ネットワークを使用するユーザ全員に対して、ユーザオブジェクトを作成します。ユーザオブジェクトは個別に管理することもできますが、次のようにすると時間の節約になります。

- ◆ テンプレートオブジェクトを使用して、通常のユーザオブジェクトのデフォルトのプロパティを設定する。新しく作成するユーザに対しては、自動的にテンプレートが適用されます (既存のユーザには適用されません)。
- ◆ 複数のユーザを一括管理するためのグループオブジェクトを作成する。
- ◆ コンテナオブジェクトをトラスティとして使用して、権利を割り当てる。これにより、権利の割り当てをコンテナ内のユーザオブジェクトすべてに適用できます。
- ◆ 複数のユーザオブジェクトは、<Shift> または <Ctrl> を押してクリックすると選択できます。これにより、選択したすべてのユーザオブジェクトのプロパティ値を一度に変更できます。

### 重要なプロパティ

ユーザオブジェクトには 80 を超えるプロパティがあります。すべてのプロパティの一覧を表示するには、iManager でユーザオブジェクトを選択します。プロパティの各ページの説明を表示するには、[ヘルプ] をクリックします。

ログイン名プロパティと姓プロパティは必須です。これら必須のプロパティおよび、他の有用なプロパティを次に示します。

- ◆ [Account Expiration Date( アカウントの有効期限 )] では、ユーザアカウントの有効期限を設定できます。有効期限を過ぎた後は、アカウントはロックされ、ユーザはログインできなくなります。

- ◆ [アカウント使用不可] には、アカウントがロックされ、ユーザがログインできない状態であることを示すシステム生成の値があります。ロックは、アカウントの有効期限が切れた場合や、ユーザが不正なパスワードを連続して何度も入力した場合などに発生します。
- ◆ [定期的なパスワード強制変更] によって、一定期間ごとにパスワードの変更をユーザに要求し、セキュリティの強化を図ることができます。
- ◆ [グループメンバーシップ] は、ユーザがメンバーとして含まれているすべてのグループオブジェクトを示します。
- ◆ [ホームディレクトリ] は、NetWare ボリュームと、ユーザ自身のファイルまでのファイルシステムのパスを表します。多くの管理者がこのようなディレクトリを作成し、ユーザの作業ファイルがネットワーク上に維持されるようにします。  
このプロパティが表すディレクトリは、ユーザオブジェクトの作成時に自動作成することもできます。
- ◆ [最終ログイン時刻] はシステムが生成するプロパティで、ユーザが最後にログインした日時を示します。
- ◆ [姓] は必須ですが、eDirectory が直接使用することはありません。eDirectory ネームベースを利用するアプリケーションが、名、役職、位置、Fax 番号など、他の識別プロパティとともに、このプロパティを使用する場合があります。
- ◆ [同時接続数の制限] では、ユーザがネットワークで開くことができるセッションの最大数を設定できます。
- ◆ [ログイン名] は iManager で [ユーザ] アイコンによって表示される名前です。また、ログイン時にユーザが入力する名前でもあります。

eDirectory では、各コンテナ内で固有のログイン名を使用する必要がありますが、ネットワーク内の別のコンテナ間では同じログイン名を使用できます。ただし、社内全体で固有のログイン名を使用した方が、管理を単純化できます。

一般に、ログイン名には姓と名の組み合わせが使用されます。たとえば、Steve Jones の場合は、STEVEJ や SJONES のようになります。

- ◆ [ログインスクリプト] では、個々のユーザオブジェクトの特定のログインコマンドを作成できます。ユーザのログイン時には、最初にコンテナのログインスクリプトが実行されます。次に、ユーザオブジェクトがプロファイルオブジェクトのメンバーシップリストに登録されている場合は、プロファイルのログインスクリプトが実行されます。最後に、ユーザのログインスクリプトが実行されます (スクリプトが存在する場合)。  
管理に要する時間を節約するため、ログインコマンドのほとんどの部分はコンテナのログインスクリプトに保管するようにします。ユーザログインスクリプトを編集すると、共通の必要条件に対する例外に対処できます。
- ◆ [ログイン時間制限] では、ユーザがログインできる時刻と曜日を設定できます。
- ◆ [ネットワークアドレス] には、ユーザのログイン元である IPX™ アドレスや IP アドレスをすべて示したシステム生成の値が格納されます。これらの値は、パケットレベルでのネットワークの問題のトラブルシューティングに役立ちます。
- ◆ [パスワード要求] では、パスワードの入力をユーザに要求するかどうかを指定できます。他の関連プロパティでは、パスワード長などの一般的なパスワード制限を設定できます。
- ◆ [ファイル/ディレクトリの権利] は、ユーザに割り当てられた、NetWare ファイルシステムに対するすべての権利を示します。iManager を使用して、ファイルおよびディレクトリに対するユーザの有効な権利 (他のオブジェクトから継承されたものを含む) を確認することもできます。



## グループ



グループオブジェクトを作成すると、複数のユーザオブジェクトを容易に管理できます。

### グループオブジェクトが表す内容

グループオブジェクトは、ユーザオブジェクトのセットを表します。

### 使用法

コンテナオブジェクトではコンテナ内のすべてのユーザオブジェクトを管理でき、グループオブジェクトでは1つまたは複数のコンテナ内のサブセットを管理できます。

グループオブジェクトは、次の2つの主な目的に対して使用されます。

- ◆ 多数のユーザオブジェクトに、一度に権利を与える。
- ◆ IF MEMBER OF 構文を使用して、ログインスクリプトコマンドを指定する。

### スタティックグループ

スタティックグループでは、メンバーオブジェクトを明示的に指定します。各メンバーは、グループに明示的に割り当てられます。

これらのグループでは静的なメンバーのリストが示され、また、グループのメンバーリストと、オブジェクト上で属性を持つメンバーとの間の参照整合性を提供します。グループメンバーシップは、メンバーの属性によって明示的に管理されます。

### ダイナミックグループ

ダイナミックグループでは、LDAP URL を使用して規則のセットを定義します。この規則に従って、eDirectory のユーザオブジェクトに一致したときに、グループのメンバーが定義されます。ダイナミックグループのメンバーは、URL に指定された検索フィルタによって定義される共通の属性を共有します。LDAP URL の形式に関する詳細は、[RFC 2255 \(http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2255.html\)](http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2255.html) を参照してください。

ダイナミックグループを使用すると、グループのメンバーシップを評価する際に使用される条件を指定できます。グループの実際のメンバーは、eDirectory によって動的に評価されます。つまり、論理的にグループ化することでグループのメンバーを定義するため、eDirectory はグループのメンバーを自動的に追加または削除できます。この拡張性の高いソリューションによって、管理コストを低減でき、LDAP の通常のグループに高い柔軟性を補うことができます。

eDirectory を使用すると、任意の属性に基づいてユーザを自動的にグループ化する場合、または一致する DN を含むグループに対して ACL を適用する場合に、ダイナミックグループを作成できます。たとえば、Department=Marketing という属性を持つすべての DN を自動的に含むグループを作成できます。Department=Marketing という検索フィルタを適用すると、Department=Marketing の属性を持つすべての DN を含むグループが検索結果として返されます。その後、このフィルタに基づく検索結果からダイナミックグループを定義できます。Department=Marketing という条件に一致するユーザがディレクトリに追加されると、このグループにも自動的に追加されます。Department が他の値に変更されたユーザ (またはディレクトリから削除されたユーザ) は、グループから自動的に削除されます。

eDirectory でダイナミックグループを作成するには、objectclass=dynamicGroup というタイプのオブジェクトを作成します。スタティックグループオブジェクトをダイナミックグループに変換するには、補助クラス dynamicGroupAux をグループオブジェクトに関連

付けます。ダイナミックグループは、グループに関連付けられた `memberQueryURL` 属性を持ちます。

`dgIdentity` 属性は、ダイナミックグループオブジェクト上で、グループのダイナミックメンバーを拡張するために使用される証明書と権利を持つエントリの識別名に設定することができます。

グループは、`memberQueryURL` を使用して管理されます。基本的な `memberQueryURL` には、ベース DN、スコープ、フィルタ、およびオプション拡張があります。ベース DN は検索ベースを指定します。スコープはベース内の検索レベルを指定します。フィルタは、指定したスコープ内で選択されたエントリに基づく検索フィルタです。

**注：** `memberQueryURL` によって作成されたリストに例外を設定するため、ダイナミックグループでもユーザを明示的に含めたり、除外したりできます。

ダイナミックグループは、Novell iManager を使用して作成および管理できます。ダイナミックグループ管理タスクには、[役割およびタスク] ページの [ダイナミックグループ] 役割をクリックしてアクセスできます。

また、LDAP コマンドを使用してもグループを管理できます。ダイナミックグループに関連付けられた最も有用なプロパティは、`dgIdentity` および `memberQueryURL` です。

### 重要なプロパティ

グループオブジェクトの最も有用なプロパティは、メンバープロパティとファイル/ディレクトリへの権利プロパティです。すべてのプロパティの一覧を表示するには、iManager でグループオブジェクトを選択します。プロパティの各ページの説明を表示するには、[ヘルプ] をクリックします。

#### ◆ `dgIdentity`

このプロパティは DN を保持します。ダイナミックグループは、この DN の識別情報を検索時の認証用に使用します。識別情報は、ダイナミックグループと同じパーティション上に存在する必要があります。`dgIdentity` によって指定されたオブジェクトは、`memberQueryURL` 属性で指定された検索を実行するのに必要な権利を持つ必要があります。

たとえば、`memberQueryURL` が次のような値であるとしします。

```
"ldap:///o=nov??sub?(title=*)"
```

この場合、`dgIdentity` は、コンテナ `o=nov` 以下の属性タイトルの読み込み / 比較権利を持っている必要があります。

#### ◆ `dgTimeout`

このプロパティは、サーバがタイムアウトになるまでにかかるメンバーの属性の読み込みまたは比較の最大所要時間を指定します。サーバがこの `dgTimeout` 値を超えると、-6016 エラーが表示されます。

#### ◆ `memberQueryURL`

このプロパティは、グループメンバーの属性と照合する規則のセットを定義します。

`memberQueryURL` は、そのスキーマ定義に従って複数の値を持つ属性です。`memberQueryURL` は複数の値を持ちますが、eDirectory 8.6.1 サーバでは、`memberQueryURL` の最初の値のみが使用されていました。

次に例を示します。

管理者によって作成されたダイナミックグループに、次のような 2 つの `memberQueryURL` 値があるとします。

```
"ldap:///o=nov??sub?cn=*"
```

```
"ldap:///o=org??sub?cn=*"
```

eDirectory 8.6.x サーバは、グループのメンバーの比較のために「ldap:///o=nov??sub?cn=\*」を使用します。複数のクエリが許可されますが、読み込まれるのは最初のクエリだけです。

この限界は、eDirectory 8.7.3 で克服されています。eDirectory 8.7.3 サーバは、すべての memberQueryURL 値に基づいてメンバーを計算するため、そのメンバーは個々の memberQueryURL 値を使用して計算されたメンバーを統合したものになります。

上の例では、結果としてダイナミックグループのメンバーは、o=org および o=nov の場合に cn 値を持つすべてのエントリとなります。

- ◆ member

このプロパティは、グループ内のすべてのオブジェクトを示します。グループオブジェクトに割り当てられた権利は、そのグループのすべてのメンバーに適用されます。ダイナミックグループの member プロパティに値を追加すると、ダイナミックグループにスタティックメンバーが追加されます。この方法は、個別にメンバーを追加する場合に使用できます。

- ◆ excludedMember

このプロパティは、ダイナミックグループのメンバーシップリストから特に除外された DN を格納します。これは、ダイナミックグループの除外リストを作成するのに使用できます。

excludedMember によって、DN をダイナミックグループのダイナミックメンバーから除外されるようにします。

これにより、memberQueryURL によって指定されたメンバーの基準で選択された場合にのみ、DN はダイナミックグループのダイナミックメンバーになり、excludedMember としてリストされたり、uniqueMember や member に明示的に追加されたりすることはありません。

- ◆ staticMember

このプロパティは、ダイナミックグループのスタティックメンバーを読み込むだけでなく、DN がダイナミックグループのスタティックメンバーかどうかを判断します。また、DN が唯一のスタティックメンバーであるダイナミックグループを探し、ダイナミックメンバーを持っているが、スタティックメンバーを持たないグループを探すこともできます。

このプロパティを既存のダイナミックグループに追加するには、dgstatic.sch を使用するスキーマを拡張します。

### eDirectory 8.6.1 以前のデータベースのダイナミックグループを更新する

ダイナミックグループがローカルで作成されるか、同期の一部として取得されると、ダイナミックグループオブジェクトが作成されますが、ダイナミックグループの機能は、そのオブジェクトに格納されるいくつかの内部値を必要とします。

以前のサーバでもダイナミックグループを格納できますが、値を生成することはできません。これは、ダイナミックグループが eDirectory 8.6.1 で導入されたためです。

eDirectory 8.6.2 では、eDirectory 8.6.1 データベースに適合させるために、8.6.1 より以前のデータベースのダイナミックグループオブジェクトへの自動更新が導入されました。

## memberQueryURL の追加構文のサポート

memberQueryURL 属性は、ダイナミックグループのメンバーを計算するために eDirectory サーバが使用する検索フィルタを格納できます。

eDirectory 8.6.1 では、フィルタで使用される属性の構文は、次の基本的な文字列型にのみ制限されていました。

- ◆ SYN\_CE\_STRING
- ◆ SYN\_CI\_STRING
- ◆ SYN\_PR\_STRING
- ◆ SYN\_NU\_STRING
- ◆ SYN\_CLASS\_NAME
- ◆ SYN\_TEL\_NUMBER
- ◆ SYN\_INTEGER
- ◆ SYN\_COUNTER
- ◆ SYN\_TIME
- ◆ SYN\_INTERVAL
- ◆ SYN\_BOOLEAN
- ◆ SYN\_DIST\_NAME
- ◆ SYN\_PO\_ADDRESS
- ◆ SYN\_CI\_LIST
- ◆ SYN\_FAX\_NUMBER
- ◆ SYN\_EMAIL\_ADDRESS


eDirectory 8.7.3 では、次の属性構文が memberQueryURL の値として追加でサポートされます。

- ◆ SYN\_PATH
- ◆ SYN\_TIMESTAMP
- ◆ SYN\_TYPED\_NAME

eDirectory 8.6.1 および eDirectory 8.7.x の両方では、SYN\_OCTET\_STRING や SYN\_NET\_ADDRESS のようなバイナリ構文は、memberQueryURL 検索フィルタでサポートされません。

詳細については、[How to Manage and Use Dynamic Groups in Novell eDirectory \(http://developer.novell.com/research/appnotes/2002/april/05/a020405.htm\)](http://developer.novell.com/research/appnotes/2002/april/05/a020405.htm) を参照してください。

## 別名

 ツリー内の別のオブジェクトをポイントする別名オブジェクトを作成できます。別名オブジェクトによって、ユーザは自分の属するコンテナの外部にあるオブジェクトに、ローカル名を付けることができます。

コンテナの名前を変更するときには、必要に応じて、元のコンテナの位置に新しい名前をポイントする別名を作成できます。これにより、コンテナ内のオブジェクトを参照するログインスクリプトコマンドやワークステーションは、コンテナ名が更新されていなくても対象のオブジェクトにアクセスできます。

## 別名オブジェクトが表す内容

別名オブジェクトは、コンテナやユーザなど、ツリー内の別のオブジェクトを表します。別名オブジェクト自体にはトラスティ権はありません。別名オブジェクトに与えられたトラスティ権は、その別名オブジェクトが示している実際のオブジェクトに適用されます。ただし、別名をトラスティ割り当ての対象とすることもできます。

## 使用法

別名オブジェクトを作成すると、名前の解決が容易になります。オブジェクトの命名規則では、現在のコンテキストのオブジェクトに対する命名が最も簡単のため、現在のコンテキストに、現在のコンテキストの外部のリソースをポイントする別名オブジェクトを作成します。

たとえば、[図 5](#) に示すように、ユーザが South コンテナにログインし、現在のコンテキストを確立する場合に、North コンテナの ColorQ というプリントキューオブジェクトにアクセスする必要があるとします。

**図 5** コンテナの例



[図 6](#) に示されるように、South コンテナに別名オブジェクトを作成できます。

**図 6** eDirectory コンテナの別名オブジェクト




別名オブジェクトによって、元の ColorQ オブジェクトがポイントされるため、South コンテナでは ColorQ をローカルオブジェクトとして印刷設定できます。

## 重要なプロパティ

別名オブジェクトには別名元オブジェクトプロパティがあり、このプロパティによって、別名オブジェクトと元のオブジェクトとが関連付けられます。

## ディレクトリマップ

 ディレクトリマップオブジェクトは、サーバのファイルシステム内のパスへのポインタです。これにより、ディレクトリをより簡単に参照できます。

ネットワークに NetWare ボリュームがない場合は、ディレクトリマップオブジェクトを作成することはできません。

### ディレクトリマップオブジェクトが表す内容

ディレクトリマップオブジェクトは、NetWare ボリューム上のディレクトリを表します (それに対し、別名オブジェクトはオブジェクトを表します)。

### 使用法

ディレクトリマップオブジェクトは、ログインスクリプトでのドライブマッピングを単純化するために作成します。ディレクトリマップオブジェクトを使用すると、複雑なファイルシステムパスを簡単な名前にすることができます。


また、ファイルの場所を変更した場合でも、新しい場所を参照するように、ログインスクリプトやバッチファイルを変更する必要がありません。ディレクトリマップオブジェクトを編集するだけです。たとえば、34 ページの  7 に示すように、South コンテナのログインスクリプトを編集するとします。

図 7 eDirectory コンテナの例



ドライブをボリューム sys: 上の Shared ディレクトリにマッピングするコマンドは、次のようになります。

```
MAP N:=sys.North.:Shared
```

共有ディレクトリマップオブジェクトを作成した場合、マップコマンドは、次のようにさらに簡単になります。


```
MAP N:=Shared
```

### 重要なプロパティ

ディレクトリマップオブジェクトには、次のようなプロパティがあります。

- ◆ 名前  
ディレクトリ内のオブジェクト (たとえば、Shared) を指定します。名前プロパティは MAP コマンドで使用されます。
- ◆ ボリューム  
Sys.North.YourCo のような、ディレクトリマップオブジェクトが参照するボリュームオブジェクトの名前が格納されます。
- ◆ パス  
public\winnt\nls\english のように、ディレクトリをボリュームのルートからのパスとして指定します。

## プロファイル

 プロファイルオブジェクトはログインスクリプトの管理に役立ちます。

### プロファイルオブジェクトが表示内容

プロファイルオブジェクトは、コンテナログインスクリプトの後、およびユーザログインスクリプトの前に実行されるログインスクリプトを表します。

### 使用法

特定のユーザのみを対象にログインスクリプトコマンドを実行したい場合は、プロファイルオブジェクトを作成します。対象のユーザには、同一コンテナ内のユーザだけでなく、異なるコンテナ内のユーザも指定できます。プロファイルオブジェクトを作成した後は、プロファイルのログインスクリプトプロパティにコマンドを指定します。続いて、該当のユーザオブジェクトをプロファイルオブジェクトのトラスティに指定し、それらのユーザオブジェクトのプロファイルメンバーシッププロパティにそのプロファイルオブジェクトを追加します。

### 重要なプロパティ

プロファイルオブジェクトには、次の2つの重要なプロパティがあります。

- ◆ ログインスクリプト

そのプロファイルのユーザに対して実行するコマンドを格納します。

- ◆ ファイル/ディレクトリへの権利

ログインスクリプトに INCLUDE ステートメントを使用した場合は、プロファイルオブジェクトに、ファイル/ディレクトリへの権利プロパティに指定されているファイルに対する権利を与える必要があります。

## コンテキストと命名規則

オブジェクトのコンテキストは、ツリー内でのそのオブジェクトの位置を表します。コンテキストは、DNS ドメインとほぼ同じです。

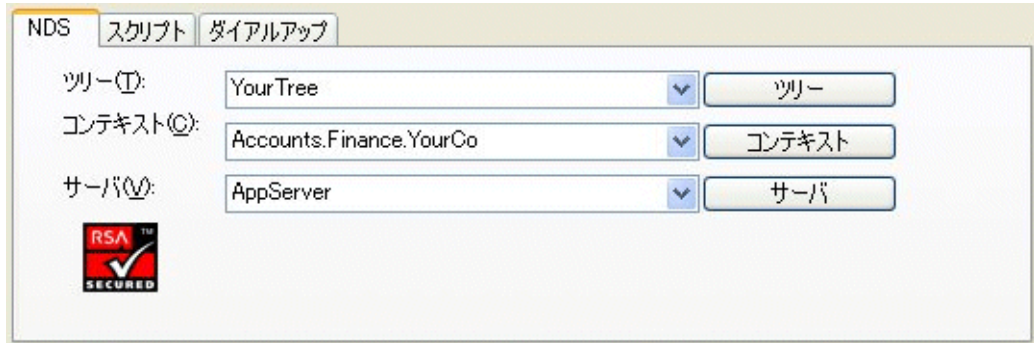
次の図は、ユーザ Bob は部門 Accounts にあり、部門 Accounts は部門 Finance にあり、部門 Finance は組織 YourCo にあることを示します。

図 8 eDirectory コンテナの例



ただし、オブジェクトのコンテキストを eDirectory ユーティリティで表現する必要がある場合もあります。たとえば、図 9 に示すように、Bob のワークステーションを設定するときには、名前前のコンテキストを指定する必要があります。

図 9 Novell Client の NDS ページ



コンテキストは、対象のオブジェクトとツリーの最上部との間に存在する各コンテナをピリオドで区切ったリストとして指定します。この例では、ユーザオブジェクト Bob はコンテナ Accounts にあり、コンテナ Accounts はコンテナ Finance にあり、コンテナ Finance はコンテナ YourCo にあります。

## 識別名

オブジェクトの識別名とは、オブジェクト名にコンテキストを付けたものです。たとえば、ユーザオブジェクト Bob の識別名は、Bob.Accounts.Finance.YourCo です。

## タイプ付きの名前

タイプ付きの名前が、eDirectory ユーティリティに表示されることがあります。タイプ付きの名前には、次の表に示されるオブジェクトタイプの略語が含まれます。

オブジェクトクラス	タイプ	略語
すべてのリーフオブジェクトクラス	共通名	CN
組織	組織	O
部門	部門	OU
国	国	C
地域	地域または都道府県	L または S

eDirectory では、タイプ付きの名前の作成に、タイプの略語、等号、およびオブジェクト名を使用します。たとえば、Bob のタイプ付きの名前は、CN=Bob となります。Bob のタイプ付きの完全な名前は、CN=Bob.OU=Accounts.OU=Finance.O=YourCo となります。eDirectory ユーティリティにおいて、タイプ付きの名前は、タイプなしの名前と互換性があります。

## ネームレゾリューション

ディレクトリツリー内のオブジェクトの位置を見つけるために eDirectory が使用するプロセスのことを、ネームレゾリューション (名前解決) といいます。eDirectory ユーティリティでオブジェクト名を使用すると、eDirectory は、現在のコンテキストまたはツリーの最上部を基準として名前の解決を行います。



## 現在のワークステーションのコンテキスト

ネットワークソフトウェアの実行時には、ワークステーションにはコンテキストが設定されます。このコンテキストによって、ネットワーク内のワークステーションの位置が相対的に指定されます。たとえば、Bob のワークステーションでは、現在のコンテキストが次のように設定されます。

Accounts.Finance.YourCo

次のセクションで説明するように、現在のコンテキストは、先頭ピリオド、相対命名、および後続ピリオドの使用法を理解するうえで重要です。

## 先頭ピリオド

現在のコンテキストがどこに設定されているかに関わりなく、ツリーの最上部から名前を解決するには、先頭ピリオドを使用します。次の例では、先頭ピリオドが CX(コンテキストの変更)ユーティリティに、ツリーの最上部を基準として名前を解決するように指定しています。

CX .Finance.YourCo

eDirectory はこのコマンドを、「ツリーの最上部から名前の解決を行い、YourCo コンテナにある Finance コンテナにコンテキストを変更する」と解釈します。

## 相対命名

相対命名とは、ツリーの最上部ではなく、ワークステーションの現在のコンテキストを基準に、名前を解決することを意味します。先頭ピリオドはツリーの最上部からの名前解決を表すため、相対命名の場合は先頭ピリオドを使用しません。

たとえば、ワークステーションの現在のコンテキストが Finance に設定されているとします。図 10 を参照してください。

図 10 eDirectory コンテナの例



Bob の相対オブジェクト名は、次のようになります。

Bob.Accounts

eDirectory は、この名前を「現在のコンテキスト Finance から解決される Accounts に属する Bob」と解釈します。

## 後続ピリオド

後続ピリオドは、相対命名でのみ使用されます。したがって、先頭ピリオドと後続ピリオドの両方を使用することはできません。後続ピリオドは、eDirectory が名前解決を開始するコンテナを変更します。

後続ピリオド 1 つにつき、解決地点がツリーの上部に向かって 1 コンテナずつ引き上げられます。たとえば、[図 11](#) の例は、ワークステーションの現在のコンテキストを Timmins から Allentown に変更する場合を示します。

図 11 eDirectory コンテナの例



この場合の適切な CX コマンドでは、次に示すように、後続ピリオドを付けた相対命名を使用します。

```
CX Allentown.East..
```

eDirectory はこのコマンドを、「現在のコンテキストの 2 つ上にあるコンテナから名前の解決を行い、East にある Allentown にコンテキストを変更する」と解釈します。

同様に、Bob が Allentown コンテナに属し、ワークステーションの現在のコンテキストが Timmins の場合、Bob の相対名は次のようになります。

```
Bob.Allentown.East..
```

## UNIX でのコンテキストと命名規則

UNIX のユーザアカウントを eDirectory に移行した場合、ユーザの命名に eDirectory のコンテキストは使用されません。ユーザのコンテキストは、UAM コンポーネントによって決定されます。

## スキーマ

「スキーマ」では、ツリー内に作成できるオブジェクトのタイプ ( ユーザ、プリンタ、グループなど )、およびオブジェクトの作成時に指定する必須情報とオプション情報が定義されます。各オブジェクトには、そのオブジェクトタイプのスキーマクラスが定義されています。

製品に元から付属しているスキーマは、ベーススキーマといいます。新しいクラスや属性の追加など、なんらかの形でベーススキーマが変更されると、そのスキーマは拡張スキーマとみなされます。

スキーマは必ずしも拡張する必要はありませんが、拡張することも可能です。iManager のスキーマ役割を使用すれば、運用条件に合わせてスキーマを拡張できます。たとえば、従業員に特殊な履き物が必要で、従業員の靴のサイズを記録しなければならない場合などに、スキーマを拡張します。この場合は、「靴のサイズ」という属性を新たに作成し、ユーザクラスに追加します。

詳細については、[4 章 105 ページの、「スキーマの管理」](#)を参照してください。

## スキーマ管理

Novell iManager のスキーマ役割によって、ツリーへのスーパーバイザ権を持っているユーザは、そのツリーのスキーマをカスタマイズできます。スキーマ役割、およびその関連タスクは、iManager の [役割およびタスク] ページに表示されています。

スキーマ役割の使用目的

- ◆ スキーマ内のすべてのクラスおよび属性の一覧を表示する。
- ◆ 構文やフラグなどの属性についての情報を表示する。
- ◆ 既存のスキーマにクラスまたは属性を追加して、スキーマを拡張する。
- ◆ 名前を付けてから、属性、フラグ、追加できるコンテナ、および属性の継承元のペアレントクラスを指定して、クラスを作成する。
- ◆ 名前を付けてから、構文およびフラグを設定して、属性を作成する。
- ◆ 既存のクラスにオプションの属性を追加する。
- ◆ 使用されていない、または必要のなくなったクラスや属性を削除する。

## スキーマクラス、属性、および構文

### クラス

クラスとは、ディレクトリオブジェクトのテンプレートのようなものです。ディレクトリオブジェクトとは、データを挿入されたクラスです。つまり、次のように表すことができます。

クラス+データ=ディレクトリオブジェクト

各クラスは、クラス名、継承クラス ( そのクラスがクラス階層の最上位である場合を除く )、クラスフラグ、および属性のグループを持っています。クラスは、ディレクトリオブジェクト ( ユーザ、プリンタ、キュー、サーバなど ) と同じように命名されますが、単なる構造であり内容はありません。

継承クラスとは、他のオブジェクトクラスを定義するときの開始点となるクラスです。継承クラスの属性はすべて、クラス階層でそのクラスの下位に位置するクラスへ継承されます。

クラス階層は、あるクラスがどのようにペアレントクラスと関連付けられているかを示します。クラス階層により、類似したクラスが関連付けられ、属性の継承が可能になります。また、クラスを格納できる有効なコンテナのタイプも定義されます。

クラスの作成時には、クラス階層と追加属性を使用して各クラスをカスタマイズできます。継承クラスを指定することによって、階層内の上位のクラスからその属性とフラグのすべてを新しいクラスに継承できます。さらに、継承された属性クラスに追加する属性を1つまたは複数選択することによって、新しいクラスをカスタマイズできます。追加の属性は、必須属性、ネーミング属性、またはオプション属性として選択できます。

オプションの属性を追加して、既存のクラスを変更することもできます。

### 属性

属性とは、eDirectory データベース内のデータフィールドのことです。たとえば、クラスが記入用紙のようなものだとすれば、属性は記入用紙における1つの記入欄です。属性は、作成時に、名前 ( 姓や社員番号など ) を付けられ、構文のタイプ ( 文字列や数字など ) が指定されます。その後、その属性はスキーママネージャの属性リストで使用できるようになります。

## 構文

構文にはいくつかの選択可能なオプションがあります。これらの構文オプションは、各属性で入力するデータのタイプを指定するために使用されます。構文は属性の作成時のみ指定できます。後から構文を変更することはできません。利用可能な構文は次のようなものです。

- ◆ バックリンク

オブジェクトを参照する他のサーバの追跡に使用されます。また、eDirectory の内部管理目的で使用されます。

- ◆ ブール

TRUE(1) または FALSE(0) の値をとる属性に使用されます。この構文タイプには、単一の値のフラグが設定されます。

- ◆ 大文字小文字を区別する文字列

比較演算で大文字 / 小文字が区別される Unicode 文字列を値としてとる属性に使用されます。2 つの「大文字小文字を区別する文字列」は、長さが等しく、対応する文字 (大文字 / 小文字の区別を含む) が同一の場合に一致とみなされます。

- ◆ 大文字小文字を無視するリスト

比較演算で大文字 / 小文字が区別されない Unicode 文字列の順序列を値としてとる属性に使用されます。2 つの「大文字小文字を無視するリスト」は、文字列の数が等しく、対応するすべての文字列が同じ場合 (つまり、長さに対応する文字が同一の場合) に一致とみなされます。

- ◆ 大文字小文字を無視する文字列

比較演算で大文字 / 小文字が区別されない Unicode 文字列を値としてとる属性に使用されます。2 つの「大文字小文字を無視する文字列」は、長さが等しく、対応する文字があらゆる面で (ただし、大文字 / 小文字の区別を除く) 同一の場合に一致とみなされます。

- ◆ クラス名

オブジェクトクラスの名前を値としてとる属性に使用されます。2 つのクラス名は、長さが等しく、対応する文字があらゆる面で (ただし、大文字 / 小文字の区別を除く) 同一の場合に一致とみなされます。

- ◆ カウンタ

増分変更された符号付き整数を値としてとる属性に使用されます。カウンタによって定義された属性は、単一の値の属性です。カウンタは、この構文の属性に値が追加されると加算で合計に追加され、この構文の属性から値が削除されると減算で合計から差し引かれるという点で、整数とは異なります。

- ◆ 識別名

eDirectory ツリー内のオブジェクト名を値としてとる属性に使用されます。DN(識別名) では大文字 / 小文字が区別されませんが、ネーミング属性の 1 つでは大文字 / 小文字が区別されます。

- ◆ 電子メールアドレス

バイナリ情報の文字列を値としてとる属性に使用されます。eDirectory は、この構文の内容の内部構造については想定していません。

- ◆ Fax 番号

国際電話番号と、推奨 T.20 に従って形式設定されたオプションのビット文字列の格納を規定する E.123 標準に準拠した文字列を指定します。2 つの Fax 番号値は、長さが等しく、対応する文字列が同一 (ただし、比較処理で無視されるスペースとハイフンを除く) の場合に、一致とみなされます。

- ◆ 保持

符号付き整数の値を持つアカウント数量の属性に使用されます。この構文は、アカウント数量 (トランザクションが完了するまでの間、サブジェクトのクレジット限度に対して暫定的に保持される金額) を表します。保持量は、カウンタ構文と同じように扱われ、新しい値がベース合計に加算されるか、ベース合計から減算されます。計算された保持量が 0 になると、保持レコードは削除されます。

- ◆ 整数

符号付き数値として表される属性に使用されます。2 つの整数値は、値が同一である場合に一致とみなされます。順序付けの比較では、符号付き整数ルールが使用されます。

- ◆ 間隔

符号付き整数を値としてとり、時間の間隔を表す属性に使用されます。間隔構文では、整数構文と同じ表現が使用されます。間隔値は、時間間隔での秒数を表します。

- ◆ ネットアドレス

サーバ環境でのネットワークレイヤアドレスを表します。このアドレスは、バイナリ形式です。2 つのネットアドレスは、各アドレスのタイプ、長さ、および値が一致する場合に、一致とみなされます。

- ◆ 数値文字列

CCITT X.208 定義で数値文字列として定義されている数値整数を値としてとる属性に使用されます。2 つの数値文字列は、長さが等しく、対応する文字が同一の場合に一致とみなされます。数値文字列文字セットにおいて有効な文字は、数字 (0 ~ 9) およびスペースのみです。

- ◆ オブジェクト ACL

アクセス制御リスト (ACL) エントリを表す値をとる属性に使用されます。オブジェクト ACL 値は、オブジェクトを保護する場合と、属性を保護場合があります。

- ◆ オクテットリスト

バイナリ情報またはオクテット文字列の順序付き文字列シーケンスを表します。オクテットリストは、保存済みリストのサブセットである場合に、保存済みリストと一致するとみなされます。2 つのオクテットリストは、長さが等しく、対応するビット列 (オクテット) が同一の場合に一致とみなされます。

- ◆ オクテット文字列

eDirectory によって解釈されないバイナリ情報の文字列を値としてとる属性に使用されます。これらのオクテット文字列は、非 Unicode 文字列です。2 つのオクテット文字列は、長さが等しく、対応するビット列 (オクテット) が同一の場合に一致とみなされます。

- ◆ パス

ファイルシステムパスを表す属性であり、サーバ上でファイルを見つけるために必要なすべての情報が格納されます。2 つのパスは、長さが等しく、対応する文字 (大文字 / 小文字の区別を含む) が同一の場合に、一致とみなされます。

- ◆ 住所

住所を表す **Unicode** 文字列を値としてとる属性に使用されます。通常、住所の属性値は、推奨 F.401 に従って **MHS Unformatted Postal O/R Address Specification** バージョン 1 から選択された属性で構成されます。この値は、30 文字 6 行 (国名を含む) に制限されます。2 つの住所は、文字列の数が等しく、対応するすべての文字列が同じ場合 (つまり、長さに対応する文字が同一の場合) に一致とみなされます。

- ◆ 印刷可能文字列

CCITT X.208 に定義されている印刷可能文字列を値としてとる属性に使用されます。印刷可能文字セットは、次のものから構成されます。

- ◆ 英大文字および英小文字
- ◆ 数字 (0 ~ 9)
- ◆ スペース
- ◆ アポストロフィ (')
- ◆ 左カッコおよび右カッコ ( )
- ◆ プラス記号 (+)
- ◆ コンマ (,)
- ◆ ハイフン (-)
- ◆ ピリオド (.)
- ◆ スラッシュ (/)
- ◆ コロン (:)
- ◆ 等号 (=)
- ◆ 疑問符 (?)

2 つの印刷可能文字列は長さが等しく、対応する文字が同一の場合に一致とみなされます。大文字 / 小文字は区別されます。

- ◆ レプリカポインタ

パーティションレプリカを表す値をとる属性に使用されます。eDirectory ツリーのパーティションでは、複数のサーバにレプリカを置くことができます。この構文は、次の 6 つのコンポーネントで構成されます。

- ◆ サーバ名
- ◆ レプリカタイプ (マスタ、セカンダリ、読み込み専用、サブオーディネートリファレンス)
- ◆ レプリカ番号
- ◆ レプリカルート ID
- ◆ アドレスの数
- ◆ アドレスレコード

- ◆ ストリーム

任意のバイナリ情報を表します。ストリーム構文を使用すると、ファイルサーバ上のファイルから eDirectory 属性を作成できます。この構文は、ログインスクリプトやその他のストリーム属性で使用されます。ストリームファイルに格納されているデータには、構文は一切適用されません。この構文は単に任意のデータであり、個別のアプリケーションで作成、定義および使用されます。

- ◆ 電話番号

電話番号を値としてとる属性に使用されます。電話番号は、1～32文字の長さの文字列です。2つの電話番号は、長さが等しく、対応する文字列が同一（ただし、比較処理で無視されるスペースとハイフンを除く）の場合に、一致とみなされます。

- ◆ 時間

符号なし整数を値としてとり、秒単位の時間を表す属性に使用されます。

- ◆ タイムスタンプ

特定のイベントが発生した時刻を示す値をとる属性に使用されます。特定のイベントが発生すると、eDirectory サーバによってタイムスタンプ値が生成され、そのイベントに関連付けられます。各タイムスタンプ値は、eDirectory パーティション内で固有です。これにより、同一パーティションのレプリカを保持しているすべてのサーバで発生したイベントを1つにまとめて順序付けできます。

- ◆ タイプ付きの名前

オブジェクトに関連付けられたレベルおよび間隔を表す値をとる属性に使用されます。この構文は、eDirectory オブジェクトを指定し、次の2つの数値を当該オブジェクトに付加します。

- ◆ 優先番号を示す属性レベル
- ◆ イベント間の秒数、または参照の頻度を表す間隔

- ◆ 不明

スキーマから削除されている属性定義をもつ属性に使用されます。この構文は、バイナリ情報の文字列を表します。

## 必須属性およびオプション属性について

オブジェクトにはどれも、オブジェクトのタイプに合わせて定義されたスキーマクラスがあります。クラスとは、意味のある方法で組織された属性のグループを指します。これらの属性の一部は必須で、一部はオプションです。

### 必須属性

必須属性とは、オブジェクトの作成時に指定する必要がある属性です。たとえば、社員番号が必須属性であるユーザクラスで新しいユーザを作成する場合、社員番号を入力せずに新しいユーザオブジェクトを作成することはできません。

### オプション属性

オプション属性とは、必要に応じて指定できる属性を指します。たとえば、ニックネームがオプション属性となっているユーザクラスで新しいユーザオブジェクトを作成するとします。この場合、この属性が入力されてもされなくてもユーザオブジェクトは作成できます。属性を入力するかどうかは、新しいユーザにニックネームが付けられているかどうかによって左右されます。

例外的にオプション属性が命名に使用される場合がありますが、その場合はこの属性は必須になります。

## スキーマの例

図 12 は、スキーマの一部の例で、ベーススキーマに類似しているかもしれません。この図は、組織クラスについての情報を表しています。この画面に表示されているほとんどの情報は、クラスが作成されたときに指定されたものです。オプションの属性のいくつかは後に追加されました。


 このアイコンは、ベーススキーマの拡張部分であるすべてのクラスおよび属性に割り当てられます。

図 12 iManager のクラス情報ページ



## スキーマを設計する

最初にスキーマの設計を行うと、長期的に見た場合、時間と労力を節約できます。ベーススキーマを表示して、それが実際の必要条件に見合うか、あるいは変更が必要かを判断できます。変更が必要な場合は、スキーママネージャを使用してスキーマを拡張します。詳細については、106 ページの「スキーマの拡張」および 109 ページの「スキーマの表示」を参照してください。

## パーティション

WAN リンクが遅い場合や信頼性に乏しい場合、またはディレクトリに多量のオブジェクトがあるためにサーバが処理しきれず、アクセスが遅くなる場合は、ディレクトリをパーティション分割することを検討してください。パーティションの詳細な説明については、5 章 117 ページの、「パーティションおよびレプリカの管理」を参照してください。

パーティションを分割すると、ディレクトリの一部を 1 つのサーバから切り離し、別のサーバに置くことができます。

パーティションは、eDirectory データベースの論理区分です。各ディレクトリパーティションは、ディレクトリ情報を格納するツリー内の個別のデータユニットとなります。



各ディレクトリパーティションは、コンテナオブジェクト、それに含まれるすべてのオブジェクト、およびそれらのオブジェクトに関するデータのセットから構成されます。eDirectory パーティションには、ファイルシステムに関する情報、またはパーティションに含まれるディレクトリやファイルに関する情報は格納されません。

パーティション分割は、Novell iManager を使用して行います。iManager では、パーティションは、パーティションアイコン (📁) によって識別されます。

図 13 サーバのレプリカビュー



この上の例では、パーティションアイコンはツリーオブジェクトの横にあります。この場合は、ツリーオブジェクトが、パーティション内の最上位のコンテナであることを意味します。他のコンテナではパーティションが表示されていないため、このパーティションが唯一のパーティションになります。

これはデフォルトの eDirectory パーティションで、ディレクトリ全体を 1 つのパーティションにまとめています。

この例では、Server1 のレプリカビューが表示されていることに注意してください。iManager でサーバのレプリカビューを表示すると、そのサーバに保持されているすべてのレプリカが右に表示されます。この場合、Server1 には唯一のパーティションのレプリカが保持されています。詳細については、[48 ページの「レプリカ」](#)および [126 ページの「eDirectory サーバのレプリカを表示する」](#)を参照してください。

## パーティション

パーティションには、その最上位のコンテナの名前が付けられます。図 14 では、Tree および Finance という名前の 2 つのパーティションがあります。パーティション Finance は Tree から分割されたため、このパーティションは Tree のチャイルドパーティションと呼ばれます。また Tree は、Finance のペアレントパーティションと呼ばれます。

図 14 パーティションのレプリカビュー



ディレクトリに大量のオブジェクトがあるためにサーバが処理しきれず、eDirectory へのアクセスが遅くなる場合、このように新しいパーティションを作成すると便利です。新しいパーティションを作成すると、データベースを分割し、その分岐内のオブジェクトを異なるサーバに渡すことができます。

上の例では、Finance パーティションのレプリカビューが表示されています。iManager でパーティションのレプリカビューを表示すると、そのパーティションのレプリカを持つサーバはすべて右側に表示されます。この場合、Server1 には、Finance パーティションの読み書き可能レプリカが保持されています。詳細については、128 ページの「パーティションレプリカを表示する」を参照してください。

## パフォーマンス向上のためにレプリカを分散する

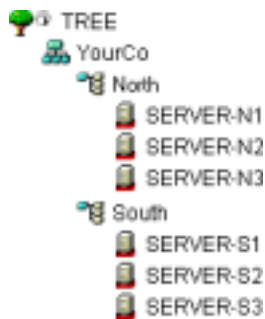
前の例で、Server1 サーバに、Tree パーティションと Finance パーティションの両方のレプリカが保持されているとします。この場合、パーティション分割後も Server1 はディレクトリ全体 ( 両パーティションのレプリカ ) をまだ保持しているため、パーティション分割による eDirectory のパフォーマンスの向上は実現されません。

パフォーマンスを向上させるには、レプリカの 1 つを別のサーバに移動する必要があります。たとえば、Tree パーティションを Server2 へ移動すると、Tree および YourCo コンテナにあるすべてのオブジェクトが Server2 に移動します。Server1 は、Finance および Accounts コンテナのオブジェクトだけを格納することになります。Server1 と Server2 への負荷は、パーティション分割を行っていないときに比べて、いずれも軽くなります。

## パーティションと WAN リンク

ネットワークが、WAN リンクで分割された North サイトと South サイトの 2 つのサイトに渡っている場合を考えてみます。各サイトには、それぞれ 3 つのサーバがあります。

図 15 eDirectory コンテナの例



このケースでは、ディレクトリを 2 つのパーティションに分割すると、eDirectory はより高速になり、信頼性も向上します。

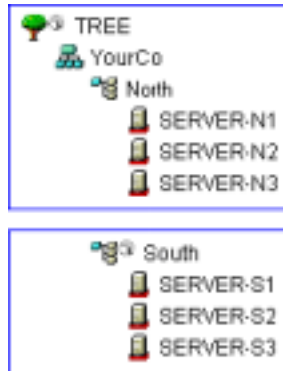
パーティションが 1 つしかない、レプリカは 1 つのサイトに保存されるか、2 つのサイトに分散されます。この場合、次の 2 つの問題点があります。

- ◆ たとえば、North サイトにすべてのレプリカが保存されているとすると、South サイトでは、ログインやリソースへのアクセスに時間がかかります。また、リンクが停止した場合、South サイトのユーザは、ログインまたはリソースへのアクセスができなくなります。

- ◆ レプリカが2つのサイトに分散されていると、ユーザはローカルでディレクトリにアクセスできます。ただし、サーバ間のレプリカの同期は WAN リンクを通して行われるので、リンクの信頼性が低い場合には、eDirectory エラーが発生する可能性があります。ディレクトリに加えられた変更を WAN 経由で伝えるには、時間がかかります。

次の図で示した、2つのパーティションによるソリューションは、WAN リンクでのパフォーマンスと信頼性の問題を解決します。

図 16 パーティションの例



ツリーパーティションのレプリカは、North サイトのサーバに保存されています。South パーティションのレプリカは、図 17 に示すように、South サイトのサーバに保存されています。

図 17 パーティション、サーバ、およびレプリカの例

Partition	Server	Replica Type
TREE	SERVER-N1	Master
	SERVER-N2	Read/write
	SERVER-N3	Read/write
South	SERVER-S1	Master
	SERVER-S2	Read/write
	SERVER-S3	Read/write

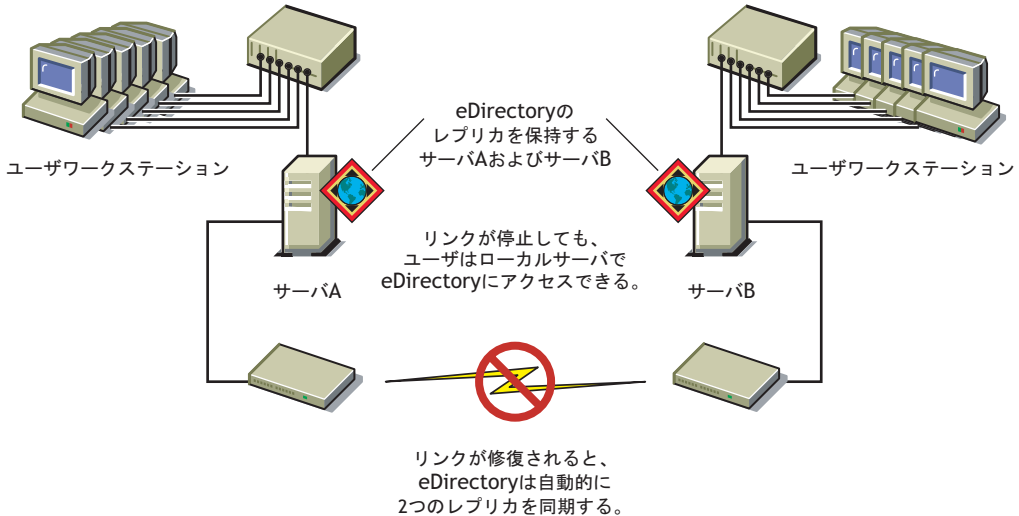
各サイトでは、ローカルリソースを表すオブジェクトは、ローカルで保存されています。サーバ間の同期トラフィックは、通信速度が遅く信頼性の低い WAN リンクではなく、LAN 上でローカルに行われます。

ただし、異なるサイトのオブジェクトにユーザや管理者がアクセスする場合は、WAN リンク上での eDirectory トラフィックが発生します。

# レプリカ

レプリカは、eDirectory サーバに分配されたユーザ定義済みのパーティションのコピーまたはインスタンスです。ネットワークに複数の eDirectory サーバがある場合、ディレクトリのレプリカ (コピー) を複数保存しておくことができます。これにより、あるサーバやそのサーバに対するネットワークリンクが機能しなくなった場合でも、ユーザが残りのネットワークリソースにログインして使用できます ( 図 18 を参照 )。

図 18 eDirectory レプリカ



それぞれのサーバは、65,000 以上の eDirectory レプリカを格納できますが、同じユーザ定義済みのパーティションのレプリカで、同じサーバ上に格納できるのは 1 つだけです。レプリカの詳細な説明については、5 章 117 ページの、「パーティションおよびレプリカの管理」を参照してください。

eDirectory の障害対策として、レプリカを 3 つ保存しておくことをお勧めします (レプリカを保存する eDirectory サーバが 3 箇所あると想定した場合)。単独のサーバに、複数のパーティションのレプリカを保存することもできます。

レプリカサーバは、eDirectory レプリカのみを格納する専用サーバです。このタイプのサーバは、DSMASTER サーバとも呼ばれます。この環境設定は、多くの単一サーバリモートオフィスを使用する企業などでよく利用されます。レプリカサーバを使用すると、リモートオフィスの場所のパーティション用として追加のレプリカを格納できます。(394 ページの「DSMASTER サーバによる災害対策」で説明されているように、それは障害回復計画の一部にもなります)。

eDirectory のレプリケーションでは、サーバファイルシステムの障害対策は提供されません。eDirectory オブジェクトに関する情報のみが、複製されます。ファイルシステムの障害対策には、TTS™(Transaction Tracking System™)、ディスクのミラーリング / 二重化、RAID、または NRS(Novell Replication Services) を使用します。

バインダリサービスを提供する NetWare サーバでは、マスタレプリカ、または読み書き可能レプリカが必要になります。

ユーザが WAN 経由で eDirectory 情報に定期的アクセスしている場合は、必要な情報を含んだレプリカをサーバに配置し、ユーザがローカルでアクセスできるようにすることで、アクセス時間と WAN トラフィックを減らすことができます。

LAN においても、ある程度は同じことが当てはまります。ネットワーク上の複数のサーバにレプリカを分散すると、情報は、通常、最も近くにある使用可能なサーバから取得されます。

## レプリカのタイプ


eDirectory では、次の図に示したレプリカのタイプがサポートされます。

図 19 レプリカのタイプ



- ◆ 49 ページの「マスタレプリカ」
- ◆ 50 ページの「読み書き可能レプリカ」
- ◆ 50 ページの「読み込み専用レプリカ」
- ◆ 50 ページの「フィルタ済み読み書き可能レプリカ」
- ◆ 51 ページの「フィルタ済み読み込み専用レプリカ」
- ◆ 51 ページの「サブオーディネートリファレンスレプリカ」

### マスタレプリカ

 マスタレプリカはオブジェクトやパーティションへの変更を開始するのに使用される、書き込み可能なレプリカタイプです。マスタレプリカは、次のタイプの eDirectory パーティション操作を管理します。

- ◆ レプリカのサーバへの追加
- ◆ レプリカのサーバからの削除
- ◆ eDirectory ツリーの新しいパーティションの作成
- ◆ eDirectory ツリーの既存のパーティションの削除
- ◆ eDirectory ツリーのパーティションの移動

マスタレプリカは、次のタイプの eDirectory オブジェクト操作にも使用されます。

- ◆ eDirectory ツリーへの新しいオブジェクトの追加
- ◆ eDirectory ツリーの既存のオブジェクトの削除、リネーム、または移動
- ◆ eDirectory ツリーへのオブジェクトの認証
- ◆ eDirectory ツリーへの新しいオブジェクト属性の追加
- ◆ 既存の属性の変更または削除

デフォルトでは、ネットワークの最初の eDirectory サーバが、マスタレプリカを保持します。マスタレプリカは、各パーティションに同時に 1 つだけ存在します。別のレプリカを作成すると、デフォルトで読み書き可能レプリカになります。

マスタレプリカを保持しているサーバを 1～2 日以上ダウンさせる場合は、読み書き可能レプリカのうち 1 つをマスタレプリカにすることができます。オリジナルのマスタレプリカは、自動的に読み書き可能レプリカになります。

ネットワーク上にマスタレプリカがない場合、eDirectory で新しいレプリカやパーティションの作成などの操作を行うことはできません。

## 読み書き可能レプリカ



eDirectory では、読み書き可能レプリカおよびマスタレプリカのオブジェクト情報にアクセスして、それを変更できます。すべての変更内容は、すべてのレプリカに自動的に伝達されます。

WAN リンクの通信速度が遅い場合やルータが使用中の場合など、ネットワークインフラストラクチャに遅延が起きてユーザへの eDirectory の反応が遅い場合は、必要なユーザの近くに読み書き可能レプリカを作成できます。読み書き可能レプリカは、サーバにいくつでも作成できますが、レプリカが増えると、同期を取るためにトラフィックの量が増加します。

## 読み込み専用レプリカ



読み込み専用レプリカは、パーティションの境界ですべてのオブジェクトに関する情報を読み込むのに使用される、読み込み可能なレプリカタイプです。読み込み専用レプリカは、マスタレプリカと読み書き可能レプリカからの同期更新は受け入れますが、クライアントからの直接の変更は受け入れません。

このレプリカタイプは、バインダリエミュレーションを提供することはできませんが、eDirectory ツリーの障害対策を提供しています。マスタレプリカおよびすべての読み書き可能なレプリカが破壊または破損された場合は、読み込み専用レプリカを新しいマスタレプリカに昇格させることができます。

また、NDS オブジェクトの読み込みやパーティションの境界内のすべてのオブジェクトを含む障害対策、およびパーティションルートオブジェクトを含む NDS ディレクトリ ツリーへの接続も提供します。

クライアントは常に読み書き可能レプリカにアクセスし、変更を加え続けることができるため、ツリー内のセキュリティポリシーを確立してオブジェクトの変更を制限する目的で読み込み専用レプリカを使用することは避けてください。権利継承フィルタの使用のように、この目的のためにディレクトリに存在するメカニズムは他にもあります。詳細については、[59 ページの「IRF\(権利継承フィルタ\)」](#)を参照してください。

## フィルタ済み読み書き可能レプリカ




フィルタ済み読み書き可能レプリカには、フィルタ済みオブジェクトセットまたはオブジェクトクラスが、これらのオブジェクトのフィルタ済み属性セットおよび値とともに格納されます。このレプリカの内容は、ホストサーバのレプリケーションフィルタに特定された eDirectory オブジェクトおよびプロパティのタイプに制限されます。ユーザはレプリカの内容を読み取ったり、変更することができ、eDirectory は選択されたオブジェクト情報にアクセスしたり、変更することができます。選択された変更内容は、すべてのレプリカに自動的に伝達されます。

フィルタ済みレプリカの場合、サーバごとにフィルタを1つだけ指定できます。つまり、あるサーバで定義されているフィルタは、そのサーバ上のすべてのフィルタ済みレプリカに適用されます。フィルタ済みレプリカは、サーバにいくつでも作成できますが、レプリカが増えると、同期を取るためのトラフィックの量が増加します。

詳細については、[51 ページの「フィルタ済みレプリカ」](#)を参照してください。

## フィルタ済み読み込み専用レプリカ

 フィルタ済み読み込み専用レプリカには、フィルタ済みオブジェクトセットまたはオブジェクトクラスが、これらのオブジェクトのフィルタ済み属性セットおよび値とともに格納されます。フィルタ済み読み込み専用レプリカは、マスタレプリカと読み書き可能レプリカからの同期更新は受け入れますが、クライアントからの直接の変更は受け入れません。ユーザはこのレプリカの内容を読み込むことができますが、変更はできません。このレプリカの内容は、ホストサーバのレプリケーションフィルタに特定された eDirectory オブジェクトおよびプロパティのタイプに制限されます。

詳細については、[51 ページの「フィルタ済みレプリカ」](#)を参照してください。

## サブオーディネートリファレンスレプリカ

サブオーディネートリファレンスレプリカは、システム生成のレプリカで、マスタレプリカや読み書き可能レプリカのオブジェクトをすべて含んでいるわけではありません。したがって、サブオーディネートリファレンスレプリカは、障害対策を提供していません。サブオーディネートリファレンスレプリカは、eDirectory によるパーティションの境界を超えた名前の解決に必要な情報のみを格納するために生成される内部ポインタです。

サブオーディネートリファレンスレプリカを削除することはできません。不要になった時点で、eDirectory が自動的に削除します。サブオーディネートリファレンスレプリカは、ペアレントパーティションのレプリカを保持し、チャイルドパーティションのレプリカを保持していないサーバでのみ作成されます。

チャイルドパーティションのレプリカが、ペアレントのレプリカを保持するサーバにコピーされた場合、サブオーディネートリファレンスレプリカは自動的に削除されます。

## フィルタ済みレプリカ

フィルタ済みレプリカには、フィルタ済みオブジェクトセットまたはオブジェクトクラスが、これらのオブジェクトのフィルタ済み属性セットおよび値とともに格納されます。たとえば、eDirectory ツリーのさまざまなパーティション内のユーザオブジェクトのみを格納するフィルタ済みレプリカのセットを、1 つのサーバ上に作成できます。また、ユーザオブジェクトのデータ（名、姓、電話番号など）のサブセットのみを格納するレプリカを作成することもできます。

フィルタ済みレプリカを使用して、1 つのサーバ上に eDirectory データのビューを構築できます。そのために、フィルタ済みレプリカでは、スコープとフィルタを作成できます。これにより、ツリー内の数多くのパーティションから 1 つの eDirectory サーバに、適切に定義されたデータセットを格納することが可能になります。

サーバのスコープとデータフィルタの説明は eDirectory に格納され、iManager のサーバオブジェクトを通して管理できます。

1 つまたは複数のフィルタ済みレプリカを格納するサーバは、レプリケーションフィルタを 1 つだけ保持します。そのため、そのサーバ上のすべてのフィルタ済みレプリカは、それぞれのパーティションからの情報の同じサブセットを格納します。フィルタ済みレプリカのマスタパーティションレプリカは、eDirectory 8.5 以降を実行している eDirectory サーバに保管する必要があります。

フィルタ済みレプリカは、次の目的に使用できます。

- ◆ 他のサーバから複製する必要があるデータ量が減ることによる、サーバへの同期トラフィックの削減。

- ◆ DirXML<sup>®</sup> によってフィルタリングする必要があるイベント量の削減。

DirXML の詳細については、『*DirXML Administration Guide (DirXML 管理ガイド)* (<http://www.novell.com/documentation/japanese/dirxml11a/index.html>)』を参照してください。

- ◆ ディレクトリデータベースのサイズの縮小。

レプリカの数が増えれば、その分データベースのサイズが増大します。完全なレプリカを作成するのではなく、特定のクラスのデータのみを格納するフィルタ済みレプリカを作成することによって、ローカルデータベースのサイズを小さくすることができます。

たとえば、ツリーに 10,000 のオブジェクトが含まれていても、そのうちユーザオブジェクトの占める割合がわずかであれば、10,000 すべてのオブジェクトを格納する完全なレプリカではなく、ユーザオブジェクトのみを格納するフィルタ済みレプリカを作成できます。

ローカルデータベースに格納されているデータをフィルタリングできるという特徴を別にすれば、フィルタ済みレプリカは通常の eDirectory レプリカと同じであり、必要なときはいつでも完全なレプリカに戻すことができます。

**注:** デフォルトでフィルタ済みのレプリカは、必須のフィルタとして組織および部門を持つこととなります。

フィルタ済みレプリカの設定と管理の詳細については、**125 ページの「フィルタ済みレプリカを設定し管理する」**を参照してください。

## NetWare バインダリエミュレーション

プリントサーバやバックアップソフトウェアなどの多くのアプリケーションは、NetWare 4 以前のバージョンの NetWare 用に作成されており、これらのアプリケーションでは、ネットワークアクセスやオブジェクトの操作に、eDirectory ではなく、NetWare バインダリを使用していました。

バインダリとは、ユーザ、グループ、およびボリュームといった、指定したサーバに認識されているオブジェクトの単純なデータベースです。バインダリは、サーバ固有であり、かつサーバ中心です。

NETX バインダリシエルなど、以前の NetWare クライアントソフトウェアはバインダリログイン手順を使用しており、ユーザは特定のサーバにしかログインできませんでした。複数のサーバにアクセスするには、複数のユーザアカウントを使用して、ログインを何度も行う必要がありました。

eDirectory では、バインダリ用に作成されたアプリケーションを、バインダリサービスを使用して実行できます。バインダリサービスでは、1 つ以上の eDirectory コンテキスト (最大 12) を、eDirectory サーバの仮想バインダリとして設定できます。設定したコンテキストは、サーバのバインダリコンテキストといいます。

バインダリサービスに関する重要な情報を次に示します。

- ◆ バインダリサービスを使用するには、eDirectory サーバにバインダリコンテキストを設定する必要があります。
- ◆ オブジェクトすべてが、バインダリオブジェクトにマップされるわけではありません。別名オブジェクトのように、オブジェクトの多くは、バインダリと互換性がありません。
- ◆ ほとんどのバインダリアプリケーションは、eDirectory で動作するようにアップグレードされています。アプリケーションの製造元に確認し、最新バージョンを取得してください。



- ◆ バインダリコンテキストを含む各 eDirectory サーバは、バインダリコンテキストを含むパーティションのマスタレプリカ、または読み書き可能レプリカを保持している必要があります。

## レプリカリングでのサーバの同期

複数のサーバが、同じパーティションのレプリカを保持している場合、それらのサーバはレプリカリングとみなされます。eDirectory は自動的にそれらのサーバを同期させ、すべてのレプリカでオブジェクトデータの整合性を保ちます。

次の eDirectory プロセスにより、レプリカリング内のサーバは同期されます。

- ◆ レプリカの同期  
レプリカの同期の詳細については、121 ページの「**レプリカの管理**」を参照してください。
- ◆ スキーマの同期
- ◆ リンバ
- ◆ バックリンク  
バックリンクの詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「**バックリンク処理の強制実行**」を参照してください。
- ◆ 接続管理

## リソースへのアクセス

eDirectory は、デフォルトの権利に従って基本レベルのネットワークアクセスセキュリティを構築します。アクセス制御をより強化するための方法を、次に示します。

- ◆ 権利の割り当て  
ユーザがネットワークリソースへのアクセスを試みると、システムはそのリソースに対するユーザの有効な権利を計算します。明示的なトラスティ割り当て、同等セキュリティの付与、および継承される権利に対するフィルタ処理を行うことによって、ユーザがリソースに対する適切で有効な権利を持つように設定できます。  
権利の割り当てを単純化するため、グループおよび職種オブジェクトを作成した後に、グループや職種にユーザを割り当てることができます。
- ◆ ログインセキュリティの追加  
ログインセキュリティは、デフォルトでは設定されていません。ログインセキュリティ方法の設定にはいくつかあります。ログインパスワード、ログインする場所および時間の制限、同時ログインセッションの制限、不正侵入者検出、およびログインの禁止があります。
- ◆ 役割ベース管理の設定  
特定のオブジェクトプロパティに対して管理者を設定し、これらのプロパティに対してのみ、権利を許可します。これにより、特定の責任を担う管理者を作成できます。この責任はどのコンテナオブジェクトの下位のオブジェクトへも継承できます。役割ベースの管理者は、従業員の情報やパスワードに関連したプロパティのような、特定のプロパティに対して責任を持ちます。  
役割ベースサービスの設定の手順については、『Novell iManager 2.0.x Administration Guide(Novell iManager 2.0.x 管理ガイド)』の「[Using the iManager Configuration Wizard \(http://www.novell.com/documentation/japanese/imanager20/imanager20/data/am48xf3.html\)](http://www.novell.com/documentation/japanese/imanager20/imanager20/data/am48xf3.html)」を参照してください。

管理者が役割ベースの管理アプリケーションを実行できるよう、特定のタスクに基づいて役割を定義することもできます。詳細については、[100 ページの「役割ベースサービスを設定する」](#)を参照してください。

## eDirectory での権利

ツリーを作成すると、デフォルトの権利割り当てによって、ネットワークへの汎用アクセスとセキュリティが与えられます。デフォルトの割り当てには、次のようなものがあります。

- ◆ ユーザ **Admin** はツリーの最上位に対してスーパーバイザ権を持ちます。これにより、**Admin** はディレクトリ全体を完全に制御できます。**Admin** は、**NetWare** サーバオブジェクトに対してもスーパーバイザ権を持ち、サーバ上のどのボリュームに対しても完全に制御できます。
- ◆ **[Public]** はツリーの最上位のブラウズ権を持ちます。これにより、すべてのユーザが、パブリックアクセスを通して、ツリー内のすべてのオブジェクトを表示する権利を持つこととなります。
- ◆ **NetWare** での移行、印刷アップグレード、または **Windows NT** でのユーザ移行などのアップグレードプロセスを通して作成したオブジェクトは、ほとんどの場合、適切なトラスティを割り当てられます。

## トラスティ割り当ておよびターゲット

権利の割り当てには、トラスティとターゲットとなるオブジェクトが関係します。トラスティとは、認証されているユーザまたはユーザのグループを表します。ターゲットとは、ユーザが権利を持っているネットワークリソースです。

- ◆ 別名をトラスティにする場合、権利は別名が示すオブジェクトにのみ適用されます。ただし、別名オブジェクトを明示的にターゲットにすることは可能です。
- ◆ ファイルシステム権は eDirectory 内ではなく、ファイルシステム自体に保存されていますが、**NetWare** ファイルシステム内のファイルまたはディレクトリも、ターゲットにすることができます。

注：[Public] トラスティは、オブジェクトではありません。すべてのネットワークユーザに対して与えられる、権利割り当てに関連するトラスティです。

## eDirectory での権利の概念

eDirectory での権利についてより良く理解していただくために、次に各権利の概念について説明します。

- ◆ [55 ページの「オブジェクト \( エントリ \) 権」](#)
- ◆ [55 ページの「プロパティ権」](#)
- ◆ [55 ページの「有効な権利」](#)
- ◆ [56 ページの「有効な権利を計算する方法」](#)
- ◆ [58 ページの「同等セキュリティ」](#)
- ◆ [59 ページの「ACL\( アクセス制御リスト \)」](#)
- ◆ [59 ページの「IRF\( 権利継承フィルタ \)」](#)

## オブジェクト(エントリ)権

トラスティ割り当てを作成するときには、オブジェクト権とプロパティ権を与えることができます。オブジェクト権はオブジェクト全体の操作に適用されますが、プロパティ権は一定のオブジェクトプロパティにのみ適用されます。オブジェクト権は、eDirectory データベース内でエントリが提供されるため、エントリ権と呼ばれます。

各権利について、次に説明します。

- ◆ **スーパーバイザ** には、オブジェクトおよびそのオブジェクトのすべてのプロパティに対する、すべての権利が含まれます。
- ◆ **参照** とは、ツリー内のオブジェクトを参照する権利です。オブジェクトのプロパティを参照する権利は含まれていません。
- ◆ **作成** は、ターゲットオブジェクトがコンテナの場合にのみ適用されます。トラスティがコンテナの下に新規オブジェクトを作成する権利のことで、ブラウズ権も含まれます。
- ◆ **削除** とは、ディレクトリからターゲットを削除する権利です。
- ◆ **リネーム** とは、ターゲット名を変更する権利です。

## プロパティ権

トラスティ割り当てを作成するときには、オブジェクト権とプロパティ権を与えることができます。オブジェクト権はオブジェクト全体の操作に適用されますが、プロパティ権は一定のオブジェクトプロパティにのみ適用されます。

iManager では、プロパティ権の管理用に次の 2 つのオプションが用意されています。

- ◆ [すべての属性権] 項目が選択されている場合は、すべてのプロパティを管理できます。
- ◆ 特定のプロパティが選択されている場合は、選択された個々のプロパティを管理できます。

各権利について、次に説明します。

- ◆ **スーパーバイザ** は、トラスティにプロパティの完全な制御権を与えます。
- ◆ **比較** により、トラスティは、プロパティの値と任意の値を比較できます。この権利によって検索が可能になります。検索では、TRUE または FALSE の結果のみ返されます。トラスティは、プロパティの値を実際に参照することはできません。
- ◆ **読み込み** により、トラスティはプロパティの値を参照することができます。この権利には、比較権が含まれます。
- ◆ **書き込み** により、トラスティはプロパティの値を作成、変更、および削除できます。
- ◆ **自己追加** とは、トラスティ自身をプロパティの値として追加または削除する権利です。この権利は、メンバーシップリストまたは ACL(アクセス制御リスト) などのような、値としてオブジェクト名を持つプロパティに適用されます。

## 有効な権利

ユーザは、明示的なトラスティ割り当て、継承、および同等セキュリティなどのさまざまな方法で権利を受け取ることができます。権利は、権利継承フィルタにより制限され、下位レベルのトラスティ割り当てによって変更、または取り消すこともできます。このような操作による最終的な結果、つまりユーザが実際に行使できる権利を、「有効な権利」といいます。

ユーザがいずれかの操作を実行しようとするたびに、該当のオブジェクトに対するそのユーザの有効な権利が計算されます。

## 有効な権利を計算する方法

ユーザがネットワークリソースにアクセスを試みるたびに、eDirectory は次の手順でそのターゲットリソースに対するユーザの有効な権利を計算します。

1. 計算で考慮される権利を持ったトラスティをリストします。考慮されるトラスティには次のものがあります。

- ◆ ターゲットリソースにアクセスしようとしているユーザ
- ◆ ユーザが同等セキュリティとなっているオブジェクト

2. リスト内の各トラスティに対して、次のように有効な権利を決定します。

- a. まず、ツリーの最上位に対してトラスティが継承可能な権利を持っているかどうかチェックします。

eDirectory はツリーオブジェクトのオブジェクトトラスティ (ACL) プロパティをチェックして、該当のトラスティが登録された項目があるかどうか調べます。該当のトラスティが登録された項目が見つかり、その権利が継承可能な場合、eDirectory はそれらの項目に指定された権利を、該当のトラスティの有効な権利の初期セットとして使用します。

- b. ツリー内の、ターゲットリソースが含まれている分岐を 1 レベル下に移動します。

- c. このレベルでフィルタリングされるすべての権利を削除します。

eDirectory はこのレベルの ACL をチェックし、該当のトラスティの有効な権利のタイプ (オブジェクト、すべてのプロパティ、または特定のプロパティ) と一致する IRF (権利継承フィルタ) がないか調べます。検出された場合は、eDirectory は該当のトラスティの有効な権利から、これらの IRF によって継承を阻止されるすべての権利を削除します。

たとえば、上位のレベルで、トラスティの有効な権利にすべてのプロパティに対する書き込み権の割り当てが含まれていても、このレベルの IRF によってその継承を阻止された場合、すべてのプロパティに対する書き込み権はトラスティの有効な権利から削除されます。

- d. このレベルで割り当てられた継承可能な権利があれば追加し、必要に応じて他の割り当てを無効化します。

eDirectory はこのレベルの ACL をチェックし、該当のトラスティが登録された項目があるかどうかチェックします。該当のトラスティが登録された項目が見つかり、その権利が継承可能な場合、eDirectory はそれらの項目の権利をトラスティの有効な権利にコピーし、必要に応じて他の割り当てを無効化します。

たとえば、上位のレベルで、トラスティの有効な権利に作成および削除オブジェクト権が含まれ、プロパティ権についてはまったく含まれていないときに、このレベルの ACL に、該当のトラスティに対する 0 オブジェクト権の割り当てとすべてのプロパティの書き込み権の割り当てが含まれている場合、トラスティの既存のオブジェクト権 (作成および削除) は 0 権利に置き換えられ、新たにすべてのプロパティ権が追加されます。

- e. ツリーの各レベル (ターゲットリソースのレベルを含む) で、フィルタリングと追加の手順 (c および d) を繰り返します。

- f. ターゲットリソースで割り当てられた継承不可能な権利があれば追加し、必要に応じて他の割り当てを無効化します。

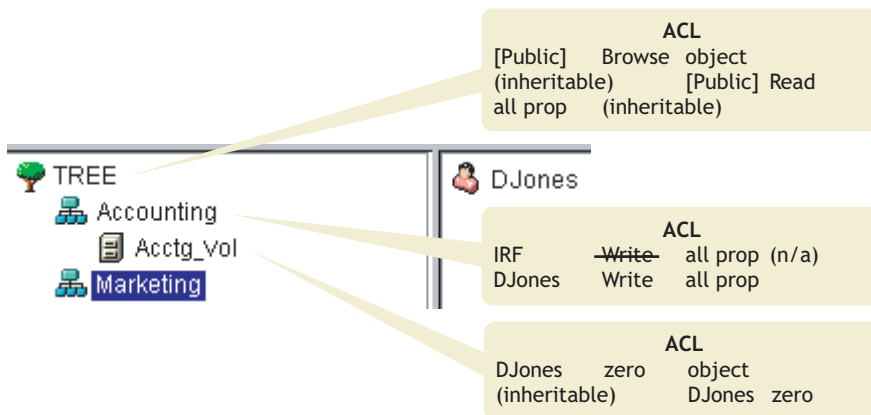
eDirectory は手順 2d と同じ処理を行います。その結果作成される権利のセットが、該当のトラスティの有効な権利になります。

3. リスト内のすべてのトラスティの有効な権利を次のように結合します。
  - a. eDirectory はリスト内のいずれかのトラスティが所有している権利をすべて含め、リスト内のどのトラスティも所有していない権利のみ除外します。権利タイプは混在させません。たとえば、特定のプロパティに対する権利を、すべてのプロパティに対する権利に追加したり、その逆を行うことはありません。
  - b. 現在有効な権利に暗黙に含まれる権利を追加します。  
権利の設定により、ターゲットリソースに対するユーザの有効な権利が作られます。

## 例

ユーザ DJones が、ボリューム Acctg\_Vol にアクセスしようとしています ( 図 20 を参照してください)。

図 20 トラスティ権の例



次の手順は、eDirectory が Acctg\_Vol に対する DJones の有効な権利を計算する方法を示したものです。

1. 計算で考慮される権利を持ったトラスティは、DJones、Marketing、Tree、および [Public] です。

ここでは、DJones がどのグループまたは役割にも所属せず、どの同等セキュリティにも明示的に割り当てられていないと仮定します。

2. 各トラスティの有効な権利は、次のとおりです。

- ◆ DJones: 0 オブジェクト、0 すべてのプロパティ

Acctg\_Vol で 0 すべてのプロパティ権を割り当てられることで、Accounting でのすべてのプロパティに対する書き込み権の割り当ては無効化されます。

- ◆ Marketing: 0 すべてのプロパティ

ツリーの最上位にあるすべてのプロパティに対する書き込み権の割り当ては、Accounting での IRF によって除外されます。

- ◆ Tree: 権利なし

Tree には、関連するどのツリー分岐でも、権利はまったく割り当てられません。

- ◆ [Public]: オブジェクトのブラウズ権、すべてのプロパティの読み込み権

これらの権利はルートで割り当てられ、関連するツリー分岐のどの位置でも、フィルタリングも無効化もされません。

3. これらすべてのトラスティの権利を結合することによって、次のようになります。

DJones: オブジェクトのブラウズ権、すべてのプロパティの読み込み権

4. すべてのプロパティの読み込み権に伴って暗黙で割り当てられるすべてのプロパティの比較権を追加することで、最終的に Acctg\_Vol に対する DJones の有効な権利は次のようになります。

DJones: オブジェクトのブラウズ権、すべてのプロパティの読み込み、比較権

### 有効な権利のブロック

有効な権利の計算方法では、IRF に頼らずに特定の権利を特定のユーザに対してブロックする方法は、必ずしも明確ではありません (IRF はすべてのユーザの権利をブロックします)。

特定の権利を IRF に頼らずにユーザに対してブロックするには、次のような方法があります。

- ◆ ターゲットリソース、およびツリー内のターゲットリソースよりも上位のレベルで、該当のユーザおよびそのユーザと同等セキュリティになるオブジェクトにそれらの権利を割り当てないようにする。
- ◆ 該当のユーザまたはそのユーザが同等セキュリティになるいずれかのオブジェクトにそれらの権利が実際に割り当てられている場合は、ツリーの下位レベルに、それらの権利の割り当てを阻止する何らかの割り当てをそのオブジェクトに対して与えるようにする。不適切な権利を持つすべての (ユーザと関連する) トラスティに対して、これを実行します。

### 同等セキュリティ

同等セキュリティとは、別のオブジェクトと同じ権利を持つことを意味します。あるオブジェクト (オブジェクト A) を別のオブジェクト (オブジェクト B) と同等セキュリティにすると、オブジェクト A の有効な権利の計算時には、オブジェクト B の権利がオブジェクト A に追加されます。

たとえば、ユーザオブジェクト Joe を Admin オブジェクトと同等セキュリティにするとします。同等セキュリティを割り当てた後は、Joe は、ツリーおよびファイルシステムに対して、Admin が持つ権利と同じ権利を持つことになります。

同等セキュリティには、次の3つのタイプがあります。

- ◆ 明示的: 割り当てによる
- ◆ 自動: グループのメンバーシップまたは役割による
- ◆ 暗黙的: すべてのペアレントコンテナおよび [Public] トラスティと同等

同等セキュリティは、1 ステップにかぎり有効です。たとえば、さらに別のユーザを上例の Joe と同等セキュリティにした場合、このユーザは Joe の権利は受け取りませんが、Admin の権利は受け取りません。

同等セキュリティは、該当のユーザオブジェクトの同等セキュリティプロパティの値として eDirectory に記録されます。

ユーザオブジェクトを職種オブジェクトにその職種の担当者として追加すると、そのユーザは自動的にその職種オブジェクトと同等セキュリティになります。ユーザをグループオブジェクトに追加した場合も同様です。

## ACL( アクセス制御リスト )

ACL( アクセス制御リスト ) は、オブジェクトトラスティプロパティとも呼ばれます。トラスティを割り当てると、そのトラスティは値としてターゲットのオブジェクトトラスティ (ACL) プロパティに追加されます。

このプロパティは、次の理由から、ネットワークのセキュリティに大きく影響します。

- オブジェクトのオブジェクトトラスティ (ACL) プロパティに対する書き込み権またはスーパーバイザ権を持つユーザは、そのオブジェクトのトラスティが誰であるかを知ることができる。
- オブジェクトのオブジェクトトラスティ (ACL) プロパティに対して自己追加権を持つユーザは、そのオブジェクトに対する自分の権利を変更できる。たとえば、そのユーザは自分にスーパーバイザ権を与えることができます。

このため、コンテナオブジェクトのすべてのプロパティに対して自己追加権を与える場合は、慎重に行う必要があります。自己追加権を割り当てられたトラスティは、該当のコンテナ、その中のすべてのオブジェクト、およびその下のコンテナ内のすべてのオブジェクトに対してスーパーバイザとなることができます。

## IRF( 権利継承フィルタ )

権利継承フィルタにより、eDirectory ツリーの低位レベルへの権利の継承をブロックできます。このフィルタの設定の詳細については、[64 ページの「eDirectory オブジェクトまたはプロパティへの権利継承をブロックする」](#)を参照してください。

## 新規サーバのデフォルト権

新規サーバオブジェクトをツリーにインストールすると、次のトラスティ割り当てが作成されます。

デフォルトトラスティ	デフォルト権
Admin( ツリー内の最初の eDirectory サーバ )	ツリーオブジェクトに対するスーパーバイザオブジェクト権。  Admin は、NetWare サーバオブジェクトに対してスーパーバイザオブジェクト権を持ちます。これは、Admin がサーバ上に存在するあらゆるボリュームのファイルシステムのルートディレクトリに対してもスーパーバイザ権を持つことを意味します。
[Public] ( ツリー内の最初の eDirectory サーバ )	ツリーオブジェクトに対するブラウザオブジェクト権。
Tree	すべてのボリュームオブジェクトのホストサーバ名プロパティおよびホストリソースプロパティに対するツリー読み込みプロパティ権。  これにより、すべてのオブジェクトが物理ボリューム名および物理サーバ名にアクセスできるようになります。
コンテナオブジェクト	sys:\public に対する読み込みおよびファイルスキャン権。これにより、コンテナの下のユーザオブジェクトは、\public の NetWare ユーティリティにアクセスできるようになります。


デフォルトトラスティ	デフォルト権
ユーザオブジェクト	ユーザ用にホームディレクトリが自動的に作成されると、ユーザにはそのディレクトリに対するスーパーバイザ権が与えられます。

## 管理の委託

eDirectory では、自分が管理するツリーの分岐に対する管理権を無効にして、その分岐の管理を他の人物に委託できます。たとえば、特別なセキュリティ要件により、ツリーの分岐を完全に制御する管理者を個別に置かなければならないような場合には、管理権の委託というこの方法をとることが必要になります。

管理権を委託するには、次を実行します。

- 1 委託先のユーザに、該当のコンテナに対するスーパーバイザオブジェクト権を与えます。

**1a** Novell iManager で、[役割およびタスク] ボタン  をクリックします。

**1b** [権利] > [トラスティの変更] の順にクリックします。


**1c** アクセスを制御するコンテナオブジェクトの名前およびコンテキストを入力して、[OK] をクリックします。

**1d** [割り当てられた権利] をクリックします。

**1e** 目的のプロパティの [スーパーバイザ] チェックボックスをオンにします。

**1f** [完了] をクリックし、[OK] をクリックします。

- 2 継承を阻止したいスーパーバイザ権や他の権利をフィルタリングするための IRF を、そのコンテナに対して作成します。

**2a** Novell iManager で、[役割およびタスク] ボタン  をクリックします。

**2b** [権利] > [権利継承フィルタの変更] の順にクリックします。

**2c** 変更する権利継承フィルタを持つオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。

**2d** 必要に応じて権利継承フィルタのリストを編集します。

フィルタのリストを編集するには、オブジェクトの ACL プロパティへのスーパーバイザ権またはアクセス制御権を持っている必要があります。オブジェクトの継承された権利を全体的にブロックするフィルタは、オブジェクトのすべてのプロパティおよび個々のプロパティに対して設定できます。

**注:** フィルタによって、このオブジェクトのトラスティに明示的に付加された権利がブロックされることはありません。これらの権利は継承されるものではないためです。

**2e** [OK] をクリックします。

**重要:** ユーザオブジェクトに管理を委託した後に、そのオブジェクトが削除されると、その分岐を管理する権利を持つオブジェクトはなくなります。

パスワードの管理など、特定の eDirectory プロパティの管理を委託するには、[62 ページの「同等セキュリティを付与する」](#)を参照してください。

役割ベース管理アプリケーションの特定の機能の使用を委託するには、[100 ページの「役割ベースサービスを設定する」](#)を参照してください。



## 権利の管理

- ◆ 61 ページの「権利を明示的に割り当てる」
- ◆ 62 ページの「同等セキュリティを付与する」
- ◆ 64 ページの「eDirectory オブジェクトまたはプロパティへの権利継承をブロックする」
- ◆ 65 ページの「eDirectory オブジェクトまたはプロパティへの有効な権利を参照する」


### 権利を明示的に割り当てる

eDirectory ツリーでデフォルトで割り当てられた権利によって、ユーザが必要以上にリソースにアクセスできたり、アクセスが不十分であったりする場合は、権利を明示的に作成して割り当てたり、それを変更したりできます。権利の割り当てを作成または変更するには、まず最初にアクセスを制御しているリソースやトラスティ(権利を所有している、またはこれから所有する eDirectory オブジェクト)を選択します。

**ヒント**：ユーザの権利を個々にではなく集団で管理するには、グループ、役割、またはコンテナオブジェクトをトラスティにします。すべてのユーザについてリソースへのアクセスを全体的に制限するには、64 ページの「eDirectory オブジェクトまたはプロパティへの権利継承をブロックする」を参照してください。

- ◆ 61 ページの「リソースに基づいて Novell eDirectory へのアクセスを制御する」
- ◆ 62 ページの「トラスティに基づいて Novell eDirectory へのアクセスを制御する」

### リソースに基づいて Novell eDirectory へのアクセスを制御する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [権利] > [トラスティの変更] の順にクリックします。
- 3 アクセスを制御する eDirectory リソース(オブジェクト)の名前およびコンテキストを指定して、[OK] をクリックします。

コンテナの下のすべてのオブジェクトへのアクセスを制御するには、そのコンテナを選択します。
- 4 トラスティのリストおよび権利の割り当てを必要に応じて編集します。
  - 4a トラスティの権利の割り当てを変更するには、トラスティを選択し、[割り当てられた権利] をクリックして、必要に応じて権利の割り当てを変更してから、[完了] をクリックします。
  - 4b トラスティとしてオブジェクトを追加するには、[トラスティの追加] をクリックし、オブジェクトを選択し、[OK] をクリックします。次に、[割り当てられた権利] をクリックしてトラスティを割り当て、[完了] をクリックします。

権利の割り当てを作成または変更する場合、オブジェクト全体に対しても、オブジェクトのすべてのプロパティまたは個々のプロパティに対しても、アクセスを付与したり拒否したりすることができます。
  - 4c トラスティとなっているオブジェクトを削除するには、そのトラスティを選択し、[トラスティの削除] をクリックします。

削除されたトラスティには、オブジェクトやプロパティに対する明示的な権利はすでにありませんが、継承や同等セキュリティによる有効な権利はまだ存在する可能性があります。
- 5 [OK] をクリックします。

## トラスティに基づいて Novell eDirectory へのアクセスを制御する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [権利] > [他のオブジェクトに対する権利] の順にクリックします。
- 3 権利を変更するトラスティ(権利を所有している、またはこれから所有するオブジェクト)の名前やコンテキストを入力します。
- 4 [検索範囲のコンテキスト] フィールドで、トラスティが現在権利を割り当てられている eDirectory オブジェクトを検索する eDirectory ツリーの一部を指定します。
- 5 [OK] をクリックします。

検索の進行状況を表す画面が表示されます。検索が終了すると、[他のオブジェクトに対する権利] ページに検索結果が表示されます。

- 6 トラスティの eDirectory 権利の割り当てを必要に応じて編集します。
  - 6a 権利の割り当てを追加するには、[オブジェクトの追加] をクリックし、アクセスを制御するオブジェクトを選択して、[OK] をクリックします。次に、[割り当てられた権利] をクリックして、トラスティの権利を割り当ててから [完了] をクリックします。
  - 6b 権利の割り当てを変更するには、アクセスを制御するオブジェクトを選択し、[割り当てられた権利] をクリックして、必要に応じてトラスティの権利の割り当てを変更してから、[完了] をクリックします。

権利の割り当てを作成または変更する場合、オブジェクト全体に対しても、オブジェクトのすべてのプロパティまたは個々のプロパティに対しても、アクセスを付与したり拒否したりすることができます。

- 6c 権利の割り当てを削除するには、アクセスを制御するオブジェクトを選択して、[オブジェクトの削除] をクリックします。

トラスティには、オブジェクトやプロパティに対する明示的な権利はすでにありませんが、継承や同等セキュリティによる有効な権利はまだ存在する可能性があります。

- 7 [OK] をクリックします。



## 同等セキュリティを付与する

別の eDirectory オブジェクトに対して同等セキュリティとなっているユーザは、事実上そのオブジェクトのすべての権利を持っています。ユーザは自動的に、所属するグループや役割に対して同等セキュリティになります。すべてのユーザは、[Public] トラスティ、およびツリーオブジェクトなど、eDirectory ツリーのユーザオブジェクトの上にある個々のコンテナに対して、暗黙で同等セキュリティとなります。また、任意の eDirectory オブジェクトに対して同等セキュリティを明示的に付与することもできます。


**注:** このセクションのタスクを実行すると、eDirectory 権利を通じて管理権を委託することができます。RBS(役割ベースサービス)役割を使用する管理アプリケーションがある場合、それらの役割にユーザメンバーシップを割り当てることで管理権を委託することもできます。

- ◆ 63 ページの「メンバーシップに基づいて同等セキュリティを付与する」
- ◆ 63 ページの「明示的に同等セキュリティを付与する」
- ◆ 64 ページの「オブジェクト固有の eDirectory プロパティの管理者を設定する」

## メンバーシップに基づいて同等セキュリティを付与する

- 1 まだ完了していない場合は、ユーザを同等セキュリティにするグループまたは役割オブジェクトを作成します。  
詳細については、92 ページの「オブジェクトを作成する」を参照してください。
- 2 グループや役割に、ユーザに必要な eDirectory 権利を与えます。  
詳細については、61 ページの「権利を明示的に割り当てる」を参照してください。
- 3 グループや役割のメンバーシップを編集して、グループや役割の権利を必要とするユーザを追加します。
  - ◆ グループオブジェクトには、[メンバー] プロパティページを使用します。  
Novell iManager で、[eDirectory 管理] > [オブジェクトの変更] の順にクリックし、グループオブジェクトの名前とコンテキストを指定して [OK] をクリックし、次に [メンバー] タブをクリックします。
  - ◆ 職種オブジェクトには、[担当者] プロパティページの [担当者] フィールドを使用します。  
Novell iManager で、[eDirectory 管理] > [オブジェクトの変更] の順にクリックし、rbsRole オブジェクトの名前とコンテキストを指定して [OK] をクリックし、次に [全般] タブで [担当者] をクリックします。
  - ◆ rbsRole オブジェクトには、[iManager のメンバーを変更] ページを使用します。  
Novell iManager で、[設定] ボタン  をクリックし、[役割の設定] > [iManager の役割を変更] の順にクリックします。次に、変更する役割の左にある [メンバーの変更] ボタン  をクリックしてから、[iManager のメンバーを変更] ページのオプションを使用して役割のメンバーを追加または削除します。
- 4 [OK] をクリックします。

## 明示的に同等セキュリティを付与する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 ユーザを同等セキュリティとするユーザまたはオブジェクトの名前およびコンテキストを入力して、[OK] をクリックします。
- 4 [セキュリティ] タブをクリックし、次のように同等セキュリティを付与します。
  - ◆ ユーザを選択し、[同等セキュリティ] をクリックします。次に、ユーザを同等セキュリティとするオブジェクトの名前とコンテキストを入力して <Enter> を押し、[OK] をクリックします。
  - ◆ ユーザを同等セキュリティとするオブジェクトを選択する場合は、[同等セキュリティ保有者] をクリックし、オブジェクトを同等セキュリティとするユーザの名前とコンテキストを入力して <Enter> を押し、[OK] をクリックします。これらの2つのプロパティページのコンテキストは、システムによって同期されます。
- 5 [OK] をクリックします。

## オブジェクト固有の eDirectory プロパティの管理者を設定する

- 1 まだ完了していない場合、オブジェクト固有のプロパティのトラスティを作成するユーザ、グループ、役割、またはコンテナオブジェクトを作成します。


トラスティとしてコンテナを作成する場合、コンテナ内またはその下のすべてのオブジェクトに権利が付与されます。プロパティは継承可能なものにする必要があります。そうしないと、コンテナおよびそのメンバーは下位レベルへの権利を持たなくなります。

詳細については、[92 ページの「オブジェクトを作成する」](#)を参照してください。

- 2 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 3 [権利] > [トラスティの変更] の順にクリックします。
- 4 管理者による管理を必要とする最高レベルのコンテナの名前とコンテキストを指定し、[OK] をクリックします。
- 5 [トラスティの変更] ページで、[トラスティの追加] をクリックし、管理者を表すオブジェクトを選択して [OK] をクリックします。
- 6 追加したトラスティの [割り当てられた権利] をクリックし、[プロパティの追加] をクリックします。
- 7 プロパティリストに追加するプロパティを選択して、[OK] をクリックします。
- 8 管理者が管理するそれぞれのプロパティについて、必要な権利を割り当てます。それぞれの権利の割り当ての [継承可能] チェックボックスをオンにします。
- 9 [完了] をクリックし、[OK] をクリックします。

## eDirectory オブジェクトまたはプロパティへの権利継承をブロックする

eDirectory では、コンテナの権利の割り当ては、継承可能な場合とそうでない場合があります。NetWare のファイルシステムでは、フォルダ上のすべての権利の割り当ては継承可能です。eDirectory および NetWare では、個々の下位アイテムについてこのような継承をブロックして、トラスティになるユーザに関係なく、これらのアイテム上では権利を無効にすることができます。例外として、NetWare ファイルシステムではスーパーバイザ権はブロックできません。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [権利] > [権利継承フィルタの変更] の順にクリックします。
- 3 変更する権利継承フィルタを持つオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。

これで、すでにオブジェクトに設定された権利継承フィルタのリストが表示されます。

- 4 プロパティページで、必要に応じて権利継承フィルタのリストを編集します。


フィルタのリストを編集するには、オブジェクトの ACL プロパティへのスーパーバイザ権またはアクセス制御権を持っている必要があります。オブジェクトの継承された権利を全体的にブロックするフィルタは、オブジェクトのすべてのプロパティおよび個々のプロパティに対して設定できます。

注：フィルタによって、このオブジェクトのトラスティに明示的に付加された権利がブロックされることはありません。これらの権利は継承されるものではないためです。

- 5 [OK] をクリックします。

## eDirectory オブジェクトまたはプロパティへの有効な権利を参照する

有効な権利は、ユーザが特定のネットワークリソース上で実行できる実際の権利です。有効な権利は、明示的な権利の割り当て、継承、および同等セキュリティを基に、eDirectory によって計算されます。システムにクエリを設定すると、リソースへのユーザの有効な権利が決定されます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [権利] > [有効な権利の表示] の順にクリックします。
- 3 参照する有効な権利を持つトラスティの名前とコンテキストを入力し、[OK] をクリックします。
- 4 次のオプションから選択します。

オプション	説明
プロパティ名	<p>トラスティが有効な権利を持つプロパティを表示します。プロパティは eDirectory から読み込まれるため、通常英語で表示されます。リストの個々のアイテムは次のタイプのいずれかです。</p> <p>[すべての属性権] - オブジェクトのすべてのプロパティを表示します。</p> <p>[エントリ権] - オブジェクト全体を表示します。スーパーバイザの場合を除き、このアイテムへの権利にはプロパティの権利は含まれません。</p> <p>特定のプロパティ-トラスティが個々に権利を持つ特定のプロパティです。デフォルトでは、このオブジェクトクラスのプロパティのみが表示されます (次を参照)。</p>
有効な権利	<p>eDirectory で計算されたとおりに、選択されたプロパティへのトラスティの有効な権利を表示します。</p>
スキーマ内のすべてのプロパティを表示する	<p>このチェックボックスをオフにすると、このオブジェクトクラスのプロパティのみが表示されます。</p> <p>eDirectory スキーマで定義されたすべてのクラスのプロパティを表示するには、このチェックボックスをオンにします。追加のプロパティが有効になるのは、このオブジェクトがコンテナである場合、またはそれが拡張されて補助クラスのプロパティを含む場合のみです。追加のプロパティの横には行頭文字は表示されません。</p>

- 5 [完了] をクリックします。



# 2

## Novell eDirectory ネットワークの設計

Novell® eDirectory™ の設計は、事実上、ネットワークのすべてのユーザおよびリソースに影響を与えます。eDirectory の設計が良ければ、ネットワークの効率性、耐障害性、安全性、拡張性、機能性が高まり、それによってネットワーク全体のパフォーマンスと価値が向上します。この章では、eDirectory ネットワーク設計上のヒントについて説明します。

- ◆ 67 ページの「eDirectory 設計の基本」
- ◆ 68 ページの「eDirectory ツリーの設計」
- ◆ 74 ページの「ツリーのパーティション化のガイドライン」
- ◆ 76 ページの「ツリーのレプリカ作成に関するガイドライン」
- ◆ 79 ページの「ユーザ環境についてのプランニング」
- ◆ 79 ページの「e ビジネスに対応する eDirectory の設計」
- ◆ 80 ページの「Novell Certificate Server について」
- ◆ 85 ページの「ネットワーク時刻の同期」

### eDirectory 設計の基本

効率のよい eDirectory 設計の基盤となるのは、ネットワークのレイアウト、会社の組織構造、および適切な準備です。

e ビジネス用に eDirectory を設計する場合は、79 ページの「e ビジネスに対応する eDirectory の設計」を参照してください。

### ネットワークのレイアウト

ネットワークのレイアウトとは、ネットワークの物理的な設定のことです。効率のよい eDirectory を設計するには、次の項目について考慮する必要があります。

- ◆ WAN リンク
- ◆ リモートアクセスを必要とするユーザ
- ◆ ネットワークリソース (サーバ数など)
- ◆ 頻繁な停電など、ネットワークの状態
- ◆ 予測されるネットワークレイアウトの変更

## 組織の構造

組織の構造は、eDirectory の設計に影響します。効率的な eDirectory を設計するには、次が必要になります。

- ◆ 組織図および運営形態についての理解
- ◆ eDirectory の設計および実装を完了するのに必要な技能のある担当者  
次の技能を持った担当者を選出する必要があります。
  - ◆ eDirectory 設計のフォーカスおよびスケジュールの管理
  - ◆ eDirectory の設計、設計標準、およびセキュリティについての理解
  - ◆ 物理的なネットワーク構造の理解および管理
  - ◆ インターネットワークのバックボーン、テレコミュニケーション、WAN 設計、およびルータ配置の管理

## eDirectory 設計の準備をする

eDirectory の設計に実際に着手する前に、次を実行します。

- ◆ スcopeおよびスケジュールの現実的な見積もりの設定
- ◆ eDirectory 実装の設計によって影響があるすべてのユーザへの連絡
- ◆ [67 ページの「ネットワークのレイアウト」](#)および [68 ページの「組織の構造」](#)の情報の参照

## eDirectory ツリーの設計

ネットワークの設計と実装で最も重要な作業は、eDirectory ツリーの設計です。ツリーの設計には、次の作業が含まれます。

- ◆ [68 ページの「命名標準ドキュメントを作成する」](#)
- ◆ [71 ページの「Tree の上位層を設計する」](#)
- ◆ [73 ページの「ツリーの下位層を設計する」](#)

## 命名標準ドキュメントを作成する

オブジェクト名などの標準名を規定すると、ユーザと管理者の両方が、ネットワークをより直感的に理解できるようになります。標準を文書化すると、管理者による、電話番号や住所などの他のプロパティ値の設定方法も指定できます。

ディレクトリの検索とブラウズでは、名前やプロパティ値の整合性が重要になります。

また、標準的な名前を使用することで、DirXML<sup>®</sup> では、eDirectory とその他のアプリケーションの間でデータを容易に移動することもできます。DirXML の詳細については、『[DirXML Administration Guide \(DirXML 管理ガイド\)](http://www.novell.com/documentation/japanese/dirxml11a/index.html) (<http://www.novell.com/documentation/japanese/dirxml11a/index.html>)』を参照してください。



## 命名規則

### オブジェクト

- ◆ コンテナ内では、重複する名前を使用することはできません。たとえば、同じコンテナ内で Debra Jones と Daniel Jones の両方に「DJONES」という名前を付けることはできません。
- ◆ 特殊文字を使用できます。ただし、プラス記号 (+)、等号 (=)、およびピリオド (.) を使用する場合は、直前に円記号 (¥) を入力する必要があります。サーバオブジェクトと国オブジェクトのほか、バイナリサービスおよび多言語環境には、追加の命名規則が適用されます。
- ◆ 大文字と小文字、およびアンダースコアとスペースは、入力時にはそのまま表示されますが、実際には区別されません。たとえば、「Manager\_Profile」と「MANAGER PROFILE」は同一と見なされます。
- ◆ 名前をコマンドラインやログインスクリプトに入力するときにスペースを使用する場合は、名前全体を引用符で囲む必要があります。

### サーバオブジェクト

- ◆ 新しいサーバをインストールすると、サーバオブジェクトが自動的に作成されます。
- ◆ 既存の NetWare® や NT サーバ、および他のツリーの eDirectory サーバに対して、追加のサーバオブジェクトを作成できますが、それらはすべてバイナリオブジェクトとして扱われます。
- ◆ サーバオブジェクトを作成するとき、その名前は物理サーバ名と一致していなければなりません。また、サーバ名には次の規定があります。
  - ◆ ネットワーク全体で固有である
  - ◆ 2～47 文字の長さである
  - ◆ A～Z までの文字、0～9 までの数字、ハイフン (-)、ピリオド (.)、およびアンダースコア ( ) のみを含む
  - ◆ 最初の文字にピリオドを使用しない
- ◆ 一度付けたサーバオブジェクト名は、iManager で変更することはできません。サーバで名前を変更すると、新しい名前が自動的に iManager に表示されます。

### 国オブジェクト

国オブジェクトは、2 文字の標準 ISO カントリーコードに従って命名します。

詳細については、ISO 3166 Code Lists (<http://www.iso.ch/iso/en/prods-services/iso3166ma/02iso-3166-code-lists/list-en1.html>) を参照してください。

### バイナリオブジェクト

NetWare 2 または NetWare 3 からバイナリサービス経由でオブジェクトにアクセスする場合は、次の制限が適用されます。

- ◆ 名前に使用されているスペースは、アンダースコアに置き換えられる。
- ◆ 名前は 47 文字で切り捨てられる。
- ◆ 次の文字は使用できない。スラッシュ (/)、円記号 (¥)、コロン (:)、コンマ (,)、アスタリスク (\*)、および疑問符 (?)。

**重要:** バイナリエミュレーションは、Linux、Solaris、AIX、HP-UX プラットフォームではサポートされていません。

## 多言語環境の注意事項

複数の言語で稼動しているワークステーションがある場合は、必要に応じて、オブジェクト名に使用する文字を、すべてのワークステーションで表示できる文字のみに制限します。たとえば、ワークステーションを日本語環境で使用している場合には、ヨーロッパの言語で表示できない文字を名前に使用しないよう制限します。

HP-UX は英語のみをサポートします。

**重要:** ツリー名は必ず英語で指定してください。

## 標準ドキュメントの例

次は、最もよく使用される一部のプロパティに関する標準ドキュメントの例です。使用しないプロパティについては、標準を規定する必要はありません。標準ドキュメントは、オブジェクトの作成または修正を担当するすべての管理者に配布します。

オブジェクトクラス   プロパティ	標準	例	理由
ユーザ   ログイン名	ファーストネームのイニシャル、ミドルネームのイニシャル (ある場合)、姓の組み合わせ (すべて小文字)。最大 8 文字です。各共通名はすべて、社内全体で固有のものにします。	msmith、bjohnson	eDirectory では会社全体で固有の名前を使用する必要はありませんが、固有にすると、同じコンテキスト (またはバインダリコンテキスト) 内の衝突を避けることができます。
ユーザ   姓	姓 (通常の大文字/小文字表記)。	Smith	メールラベルの生成に使用されます。
電話番号および Fax 番号	ハイフンで区切られた番号。	米国 :123-456-7890 その他: 44-344-123456	自動ダイヤルソフトウェアで使用されます。
複数のクラス   地域	2 文字の地域コード (大文字)、ハイフン、メール配達地点の組み合わせ。	BA-C23	社内のメール配達で使用されます。
組織   名前	すべてのツリーに与える会社名。	YourCo	独立したツリーがある場合、標準の組織名を使用することで、将来ツリーのマージができます。
部門   名前 (地域に基づく)	2 または 3 文字の地域コードで、すべて大文字。	ATL、CHI、CUP、LA、 BAT、BOS、DAL	短く、標準的な名前を使用することで、効率的に検索できます。
部門   名前 (部署に基づく)	部署名または略語。	Sales、Eng	短く、標準的な名前を使用することで、どの部署で使用しているコンテンツか見分けやすくなります。
グループ   名前	識別名。	Project Managers	ユーティリティによってはすべて表示できない場合があるため、極端に長い名前は避けます。
ディレクトリマップ   名前	ディレクトリマップが示すディレクトリの内容。	DOSAPPS	短く、標準的な名前を使用することで、どの部署で使用しているコンテンツか見分けやすくなります。
プロフィール   名前	プロフィールの目的。	MobileUser	短く、標準的な名前を使用することで、どの部署で使用しているコンテンツか見分けやすくなります。
サーバ   名前	SERV、ハイフン、部署、ハイフン、固有番号の組み合わせ。	SERV-Eng-1	eDirectory では、ツリー内で固有のサーバ名を使用する必要があります。

## Tree の上位層を設計する

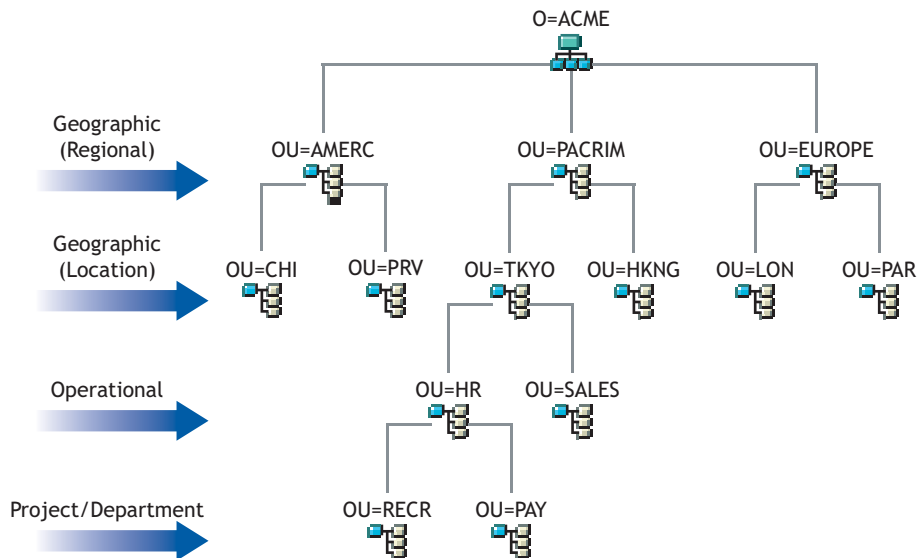
ツリーの上位層を変更するとツリーの残りの部分すべてに影響するため、ツリーの上位層は慎重に設計します。WAN リンクがある場合は、特に注意する必要があります。ツリーの最上部は、後からの変更ができるだけ必要とにならないように設計します。

eDirectory ツリーの作成時には、次の eDirectory 設計規則に従います。

- ◆ ピラミッド型的设计を使用する。
- ◆ eDirectory ツリーは 1 つにし、固有の名前を付ける。
- ◆ 組織オブジェクトを 1 つ作成する。
- ◆ 物理的なネットワーク構造を表す第 1 レベルの部門群を作成する。

71 ページの 図 21 は、eDirectory 設計規則を示しています。

図 21 eDirectory 設計規則



ツリーの上位層を作成するには、92 ページの「オブジェクトを作成する」および 92 ページの「オブジェクトのプロパティを変更する」を参照してください。

### ピラミッド型的设计の使用

ピラミッド型の eDirectory では、管理や、大きなグループに対する変更、および論理パーティションの作成をより容易に実行できます。

ピラミッド型以外の設計として、フラットツリー型があります。この場合、すべてのオブジェクトがツリーの最上部に置かれます。eDirectory をフラットツリー型に設計することも可能ですが、管理やパーティション化がより難しくなります。

### 固有の名前を持つ単一の eDirectory ツリーの使用

ほとんどの組織では、ツリーを 1 つにするのが最適です。デフォルトでは、ツリーは 1 つだけ作成されます。ツリーが 1 つだけだと、ネットワーク内のユーザの識別を統一でき、セキュリティ管理もより容易になります。また、1 箇所で集中管理できます。

業務では単一のツリーの使用を推奨しますが、追加のツリーのテストや開発を否定するわけではありません。

組織によっては、法的または政治的な問題、あるいは社内の事情から、複数のツリーが必要になる場合もあります。たとえば、自立した 7 つの組織から構成される組織では、7 つのツリーが必要になることも考えられます。組織で複数のツリーが必要な場合は、DirXML を使用して管理を容易化することを検討してください。DirXML の詳細については、『*DirXML Administration Guide(DirXML 管理ガイド)* (<http://www.novell.com/documentation/japanese/dirxml11a/index.html>)』を参照してください。

**注:** HP-UX は DirXML をサポートしていません。

ツリーには、他のツリー名と重複しない、固有の名前を付けます。「EDL-TREE」のように、短くてわかりやすい名前を指定します。

同じネットワーク内に同じ名前のツリーが存在すると、次のような問題が発生する場合があります。

- ◆ 更新内容が、対象のツリーとは別のツリーに適用される
- ◆ リソースが消失する
- ◆ 権利が消失する
- ◆ データが破損する

ツリー名は、DSMerge ユーティリティで変更できますが、変更は十分に注意して行ってください。ツリー名の変更はネットワーク全体に影響します。ツリー名を変更した場合は、新しいツリー名でクライアントを再設定する必要があります。

## 単一の組織オブジェクトの作成

通常、1 つの eDirectory ツリーには、1 つの組織オブジェクトを作成します。デフォルトでは、組織オブジェクトが 1 つ作成され、それに会社名に基づいた名前が付けられます。これにより、会社全体に適用される変更をツリー内の 1 個所から設定できます。

たとえば、ZENworks<sup>®</sup> を使用して、組織オブジェクトの下に、ワークステーションインポートポリシーオブジェクトを作成できます。このポリシーは、eDirectory 内にワークステーションオブジェクトを作成するときの命名方法を定義するもので、組織全体に影響します。

組織コンテナには、次のオブジェクトが作成されます。

- ◆ 管理者
- ◆ サーバ
- ◆ ボリューム

eDirectory が実行されている Windows、Linux、Solaris、AIX、または HP-UX サーバのみを含むネットワークには、ボリュームオブジェクトはありません。

次のようなケースでは、組織オブジェクトを複数作成することが必要になる場合があります。

- ◆ 複数の会社から構成される企業で、それぞれの会社が個別にネットワークを構成している。
- ◆ 社内で、独立した業務単位または組織を表す必要がある。
- ◆ 各部門が独立した形態をとることを規定する社内ガイドラインやポリシーがある。

## 物理的なネットワークを表す部門を作成する

第 1 レベルの部門設計は、eDirectory の効率性とパーティション化に影響するため重要です。

LAN または WAN を使用して複数のビルや場所などにまたがって構成されているネットワークでは、所在地に基づいて第 1 レベルの部門オブジェクトを設計します。これにより、1 つのパーティション内のすべてのオブジェクトが 1 箇所に維持されるように、eDirectory を分割できます。また、各場所でのセキュリティの設定や管理者の割り当ても容易になります。

## ツリーの下位層を設計する

ツリーの下位層は、ネットワークリソースの編成に基づいて設計します。eDirectory ツリーの下位層の設計は同じ場所に存在するオブジェクトにのみ影響するため、下位層は上位層よりも自由に設計できます。

ツリーの下位層を作成するには、**92 ページの「オブジェクトを作成する」**および**92 ページの「オブジェクトのプロパティを変更する」**を参照してください。

## コンテナ、ツリー、およびデータベースのサイズを決定する

作成する下位層のコンテナオブジェクトの数は、ツリー内のオブジェクトの総数、空きディスク容量、およびディスクの入出力速度制限によって決まります。eDirectory は、1 つの eDirectory ツリー内に 10 億以上のオブジェクトを格納できることがテストで確認されています。したがって、実際に制約となる項目は、空きディスク容量とディスクの入出力速度、およびパフォーマンスを維持するための RAM です。大規模なツリーの場合は、レプリカの同期処理が大きく影響することにも注意が必要です。

eDirectory 内の一般的なオブジェクトのサイズは 3 ~ 5KB です。この数値に基づいて、現在保持している、または新たに作成するすべてのオブジェクトの保管に必要な空きディスク容量をすばやく計算できます。オブジェクトのサイズは、データに付属する属性の数や、データの内容に対応して大きくなります。画像やサウンド、または生物統計学などの BLOB(バイナリラージオブジェクト) データを格納するオブジェクトの場合、そのサイズは増加します。

パーティションのサイズが大きくなるほど、レプリケーションのサイクルは遅くなります。ZENworks や DNS/DHCP サービスなど、eDirectory を使用する必要がある製品を使用する場合、これらの製品によって作成された eDirectory オブジェクトにより、格納先のコンテナのサイズが影響を受けます。場合によっては、DNS/DHCP など、管理目的のみで使用するオブジェクトは固有のパーティションに格納するように検討してください。パーティションを別にすれば、レプリカの同期処理の速度の低下によってユーザのアクセスに支障をきたすことを避けることができます。また、パーティションとレプリカの管理も容易になります。

必要な場合は、eDirectory データベースまたは DIB(ディレクトリ情報ベース) セットのサイズを簡単に判別できます。

- ◆ NetWare の場合は、**Novell Support Web サイト (<http://support.novell.com>)**から toolbox.nlm をダウンロードして、サーバ上の sys:\_netware ディレクトリを表示します。
- ◆ Windows の場合は、\novell\nds\dibfiles にある DIB セットを参照します。
- ◆ Linux、Solaris、AIX、または HP-UX の場合は、インストール時に指定したディレクトリにある DIB セットを参照します。

## 作成するコンテナを決定する

通常、同じ必要性からアクセスされる eDirectory オブジェクトをまとめて格納するコンテナを作成します。それにより、1 つのトラスティ割り当てまたはログインスクリプトで多くのユーザにサービスを提供できます。特にログインスクリプトをより効率的にすることを目的としてコンテナを作成できるほか、1 つのコンテナに 2 つの部署を割り当てることによって、ログインスクリプトの管理をより容易にすることもできます。

ネットワーク内のトラフィックを制限するため、ユーザは各自が必要とするリソースの近くに配置するようにします。たとえば、同じ部署で働く社員は、通常、隣り合った位置で作業します。それらの社員は同じファイルシステムにアクセスし、同じプリンタを使用して印刷します。

一般的なワークグループ境界の例外については、容易に対処できます。たとえば、2 つのワークグループが共通のプリンタを使用するような場合は、一方のワークグループのプリンタに対して別名オブジェクトを作成します。グループオブジェクトを作成することによって、1 つのワークグループ内だけでなく、複数のワークグループに存在する一部のユーザオブジェクトをまとめて管理できます。また、固有のログインスクリプト要件を持つ、一部のユーザ用のプロファイルオブジェクトも作成できます。

## ツリーのパーティション化のガイドライン

eDirectory にパーティションを作成すると、データベースの各部分は複数のサーバに分割して格納されます。パーティション化によって、eDirectory データ処理と保管の負荷がネットワーク内の複数のサーバに分散されるため、ネットワークの使用を最適化できます。デフォルトでは、パーティションは 1 つだけ作成されます。パーティションの詳細については、[44 ページの「パーティション」](#)を参照してください。パーティションの作成については、[5 章 117 ページの、「パーティションおよびレプリカの管理」](#)を参照してください。

次のガイドラインは、ほとんどのネットワークに適用できる一般的な規則です。特定の環境設定、ハードウェア、およびトラフィックの処理量など、それぞれのネットワークの運用条件に合わせて若干の調整が必要になる場合もあります。

## ツリーの上位層におけるパーティションを決定する

ツリーをピラミッド型に設計するのと同じように、パーティションもピラミッド型に構成します。パーティションの構造は、ツリーの最上位に少数のパーティションがあり、下位になるに従ってパーティションの数が増えていくようにします。このような設計では、下位層より上位層のほうがパーティション数が多いツリー構造に比べて、作成されるサブオーディネートリファレンスの数が少なくなります。

ツリーをピラミッド型に設計するには、常にリーフオブジェクト（特にユーザ）の比較的近くにパーティションを作成するようにします（インストール時にツリーのルートに作成されるパーティションについては例外です）。

上位層のパーティションを設計する場合は、次の点を考慮します。

- ◆ WAN インフラストラクチャに基づいて、ツリーの最上部をパーティション化する。ツリーの最上部にはパーティションを少なめに配置し、下位になるに従ってパーティションの数を増やします。

WAN リンクによって区切られている各サイトのコンテナを作成し（各サーバオブジェクトをそのローカルコンテナに格納）、その後、各サイトのパーティションを作成します。

- ◆ WAN リンクを持つネットワークでは、パーティションが複数の場所にまたがらないようにする。  
パーティションを場所ごとに作成すれば、異なるサイト間のレプリケーショントラフィックによって WAN 帯域幅が不必要に消費されることはありません。
- ◆ ローカルサーバを中心にしてパーティションを作成する。物理的に離れた場所にあるサーバは別のパーティションに格納します。

WAN トラフィック管理の詳細については、[10 章 241 ページの「WAN トラフィックマネージャ」](#)を参照してください。

## ツリーの下位層におけるパーティションを決定する

eDirectory ツリーの下位層のパーティションを設計する場合は、次を考慮します。

- ◆ 下位層のパーティションは、事業部、部署、およびワークグループと、それらに関連するリソースに基づいて定義する。
- ◆ 各パーティションについて、パーティション内のすべてのオブジェクトが同じ場所にあるようにする。それによって、eDirectory への更新がローカルサーバで行われるようになります。

## パーティションサイズを決定する

eDirectory では、パーティションのサイズについて次の設計制限を推奨しています。

エレメント	制限
パーティションのサイズ	無制限のオブジェクト レプリカ DIB(ディレクトリ情報ベース)は 1TB に制限
ツリー内のパーティションの総数	無制限
ペアレントパーティションごとのチャイルドパーティション数	150
パーティションごとのレプリカ数	50 レプリカ DIB による制限
レプリカサーバごとのレプリカ数	250

設計ガイドラインが NDS<sup>®</sup> 6 および 7 からこのように変更されたのは、NDS 8 においてアーキテクチャが変更されたためです。これらの推奨値は、企業などの分散環境に適用されます。ただし、これらは e ビジネスやアプリケーションには適用されない場合があります。

一般的な e ビジネスユーザに対しては、すべてのデータが単一のサーバに保管されることが必要ですが、eDirectory 8.7.3 では、ツリー内のさまざまなエリアに属するオブジェクトや属性の一部を格納するフィルタ済みレプリカを作成できます。この機能を利用すると、1 つのサーバにすべてのデータを保管しなくても、同じように e ビジネスに対応できます。詳細については、[51 ページの「フィルタ済みレプリカ」](#)を参照してください。

## ネットワーク変数について

パーティションを設計する場合は、次のネットワーク変数について考慮します。

- ◆ サーバの数および速度
- ◆ ネットワークアダプタ、ハブ、ルータなどのネットワークインフラストラクチャの速度
- ◆ ネットワークトラフィックの量

## ツリーのレプリカ作成に関するガイドライン

複数の eDirectory パーティションを作成するだけでは、耐障害性やディレクトリのパフォーマンスは向上しません。耐障害性やパフォーマンスの向上は、複数のレプリカを効果的に使用することで可能になります。レプリカの配置は、アクセス可能性や耐障害性にとってきわめて重要です。eDirectory のデータには、できるだけすばやくアクセスできるようにする必要があります。また、eDirectory のデータは障害対策用に複数の場所にコピーする必要があります。レプリカの作成については、[5 章 117 ページの「パーティションおよびレプリカの管理」](#)を参照してください。

レプリカの配置計画では、次のガイドラインを参考にしてください。

- ◆ [76 ページの「ワークグループのニーズ」](#)
- ◆ [76 ページの「障害対策」](#)
- ◆ [77 ページの「レプリカ数を決定する」](#)
- ◆ [77 ページの「ツリーパーティションのレプリカを作成する」](#)
- ◆ [78 ページの「管理用にレプリカを作成する」](#)
- ◆ [78 ページの「NetWare のバインダリサービスの必要性に対応させる」](#)
- ◆ [78 ページの「WAN トラフィックを管理する」](#)

## ワークグループのニーズ

各パーティションのレプリカを、そのパーティション内の情報を使用するワークグループに物理的に近いサーバ上に作成します。WAN リンクの一方の側のユーザが他方の側のサーバに格納されているレプリカに頻繁にアクセスする場合は、WAN リンクの両端のサーバ上にそれぞれレプリカを作成します。

レプリカは、ユーザ、グループ、およびサービスによるアクセスが最も多い場所に配置します。2 つの独立したコンテナ内の各ユーザグループが別のパーティション境界内の同じオブジェクトにアクセスする必要がある場合は、これらのユーザグループが属する 2 つのコンテナより 1 つ上のレベルにあるコンテナ内のサーバ上にレプリカを配置します。

## 障害対策

ディスクがクラッシュしたり、サーバがダウンした場合、別の場所にあるサーバ上のレプリカによりユーザのネットワークへの認証を行い、使用不能になったサーバに格納されているパーティション内のオブジェクトの情報を取得できます。

いくつかのサーバに同じ情報が配布されていると、ユーザのネットワークへの認証やログインなどのサービスを提供するために、1 つのサーバに依存する必要がありません。



耐障害性を確保するには、ディレクトリツリーに十分な数のサーバがあれば、各パーティションについて3つのレプリカを作成します。ローカルパーティションのローカルレプリカは、最低2つは必要です。他の場所にあるデータへのアクセスを確保する場合や、負荷分散および障害対策用にデータのインスタンスを複数維持する必要があるアプリケーションを使用している場合、またeビジネスを展開している場合を除いて、通常は、レプリカの作成は3つまでで十分です。

マスタレプリカは1つだけ保持できます。その他のレプリカは、読み書き可能、読み込み専用、またはフィルタ済みレプリカとして使用します。通常、ほとんどのレプリカは、読み書き可能レプリカとして使用されます。読み書き可能レプリカは、マスタレプリカ同様、オブジェクトの表示、オブジェクトの管理、およびユーザのログインを処理できます。変更が加えられると、読み書き可能レプリカは同期情報を送信します。

読み込み専用レプリカには、書き込みはできません。読み込み専用レプリカは、オブジェクトを検索および表示できます。また、パーティションのレプリカの同期が実行されると、更新されます。

サブオーディネートリファレンスレプリカやフィルタ済みレプリカは、障害対策用にはなりません。サブオーディネートリファレンスはポインタであり、パーティションのルートオブジェクト以外のオブジェクトは格納されません。フィルタ済みレプリカには、パーティション内の特定の一部のオブジェクトのみが格納されます。

eDirectory では、パーティションごとに数に制限なくレプリカを作成できますが、レプリカの数が増えるに従って、ネットワークトラフィックの量も増大します。障害対策とネットワークパフォーマンス、それぞれの必要性をバランスよく配慮します。

サーバ上の各パーティションには、レプリカを1つだけ格納できます。1つのサーバには、複数のパーティションのレプリカを格納できます。

障害発生によって、特定の場所のすべてのデータまたはその中のいずれかのサーバ上のデータが失われた場合、その組織の障害回復計画に基づくネットワーク再構築のほとんどの作業は、パーティションのレプリカを使用して行うことができます。サーバが1つしかない場所では、eDirectory を定期的にバックアップします (一部のバックアップソフトウェアでは、eDirectory のバックアップアップが実行されません)。障害対策用のレプリカを作成するための、追加サーバの購入を検討してください。

## レプリカ数を決定する

複数のレプリカの作成を制限する要素となるのは、それぞれのレプリカの同期で必要となる処理時間とトラフィックの量です。オブジェクトが変更されると、変更内容がレプリカリング内のすべてのレプリカに伝達されます。レプリカリング内のレプリカの数が多いほど、変更内容の同期に必要なデータの伝送量も増えます。WAN リンク経由でレプリカを同期する必要がある場合、同期にかかる時間のコストが高くなります。

地理上の多くの場所でパーティションを作成した場合、一部のサーバではサブオーディネートリファレンスレプリカが多量に生成されることとなります。eDirectory では、地域のパーティションを作成することによって、これらのサブオーディネートリファレンスをより多くのサーバに分散させることができます。

## ツリーパーティションのレプリカを作成する

ツリーパーティションはeDirectory ツリー内の最も重要なパーティションです。このパーティションの唯一のレプリカが破損した場合、そのパーティションが修復されるか、eDirectory ツリー全体が再構築されるまで、ネットワークの機能が停止します。また、ツリーに関連する設計の変更はまったくできなくなります。

Tree パーティションのレプリカの作成時には、サブオーディネートリファレンスの同期にかかるコストを考慮しながら、レプリカ数を決めてください。

## 管理用にレプリカを作成する

パーティションの変更はマスタレプリカからのみ発生するため、中央サイトのネットワーク管理者のすぐ近くにあるサーバに各マスタレプリカを保管するようにします。一見、それぞれのリモートサイトにマスタを保管するほうが適切のように思えますが、マスタレプリカはパーティションの操作が行われる場所に保管するのが妥当です。

パーティション作成などの重要な eDirectory の操作は、中央サイトの特定の管理者または管理者グループが担当することを推奨します。担当者を限定することによって、eDirectory の動作を悪化させるエラーの発生の可能性を制限できるほか、マスタレプリカを中央で一括してバックアップできます。

ネットワーク管理者は、ネットワークトラフィックが少ない時間帯を選んで、レプリカの作成などのコストの高い操作を実行します。

## NetWare のバインダリサービスの必要性に対応させる

NetWare で eDirectory を使用する場合に、ユーザがバインダリサービスを通してサーバにアクセスする必要があるときは、そのサーバにバインダリコンテキストを格納するマスタレプリカまたは読み書き可能レプリカが保管されている必要があります。バインダリコンテキストは、autoexec.ncf 内の SET BINDERY CONTEXT ステートメントによって設定されます。

ユーザは、実オブジェクトがそのサーバ上に存在する場合のみ、バインダリサービスを提供するオブジェクトにアクセスできます。パーティションのレプリカをサーバに追加すると、サーバに実オブジェクトが追加されます。それにより、そのパーティションにユーザオブジェクトを持つユーザは、バインダリ接続によってサーバにログインできるようになります。

バインダリサービスの詳細については、[52 ページの「NetWare バインダリエミュレーション」](#)を参照してください。

## WAN トラフィックを管理する

現在、ユーザが WAN リンクを使用して特定のディレクトリ情報にアクセスしている場合、必要な情報を格納するレプリカをユーザのローカルサーバに作成することによって、ユーザのアクセス時間と WAN トラフィックの量を削減できます。

マスタレプリカをリモートサイトに複製する場合や、アクセスの提供や障害対策用に WAN 経由でレプリカを配置する場合には、レプリカの同期処理で使用される帯域幅について考慮します。

推奨する 3 つのレプリカの作成がローカルサイトで不可能な場合に耐障害性を確保するためと、アクセス可能性を高めるため、またマスタレプリカを中央で一括して管理および保管するためにのみ、ローカルサイト以外の場所にレプリカを作成します。

WAN リンク上の eDirectory トラフィックのレプリケーションを制御するには、WAN マネージャを使用できます。詳細については、[10 章 241 ページの「WAN トラフィックマネージャ」](#)を参照してください。

# ユーザ環境についてのプランニング

eDirectory ツリーの基本構造を設計し、パーティションとレプリカを設定した後は、管理を単純化し、ネットワークリソースへのアクセスを容易にするためのユーザ環境の設定プランを立てます。ユーザ環境プランを立てるには、各ユーザの必要条件を検討し、エリアごとにアクセスに関するガイドラインを作成します。

## ユーザの必要条件を検討する

ユーザが必要とする条件を検討する際には、次の点について考慮します。

- ◆ プリンタやファイル保管領域など、物理的なネットワークに関する必要条件  
1つのツリー内の複数のユーザグループ、または複数のコンテナに属する複数のユーザグループによってリソースが共有されていないか確認します。また、リモートユーザが必要とする物理的なリソースについても考慮します。
- ◆ NetWare ユーザのバインダリサービスに関する必要条件  
バインダリサービスに基づくアプリケーションはどれか、またどのユーザによって使用されるか確認します。
- ◆ アプリケーションに関する必要条件  
ユーザが必要とするアプリケーションおよびデータファイルはどれか、オペレーティングシステムは何か、またアプリケーションにアクセスする必要があるユーザまたはユーザグループは誰かを確認します。共用アプリケーションを、ZENworksなどのアプリケーションによって自動で起動するのか、または手動で起動するのか確認します。

## アクセスに関するガイドラインを作成する

ユーザの必要条件についての情報を収集した後は、ユーザ環境の構成に使用する eDirectory オブジェクトを決定します。たとえば、ポリシーパッケージやアプリケーションオブジェクトを作成する場合は、作成数、およびツリー内の保管先について決定します。

また、ユーザのアクセスを制限するセキュリティの実装方法についても決める必要があります。個々のセキュリティ項目に関連する、セキュリティ上の予防措置がないか検討します。たとえば、eDirectory スーパーバイザ権はファイルシステムによって継承されるため、サーバオブジェクトに対してはスーパーバイザ権を付与しないよう、ネットワーク管理者に警告する場合があります。

## e ビジネスに対応する eDirectory の設計

eDirectory を e ビジネスに利用する場合 ( サービスのポータルを提供したり、他の企業とデータを共有する場合など )、この章ですでに説明している推奨事項は適用されないことがあります。

その場合は、e ビジネスに対応する eDirectory の設計のガイドラインとして次に示す推奨事項を代わりに使用できます。

- ◆ コンテナ数を制限したツリーを作成する。  
このガイドラインは、使用するアプリケーション、および eDirectory の実装法に応じて適用します。たとえば、メッセージサーバをグローバルに展開する際には、この章の前半で説明した従来の eDirectory 設計ガイドラインを適用したほうがより適

切な場合があります。また、ユーザの管理を分散させたい場合は、管理責任のエリアごとに独立した OU(部門)を作成することが必要になる場合があります。

- ◆ 最低2つのパーティションを維持する。

ツリーレベルに作成されるデフォルトのパーティションを維持し、その他にツリーの残りの部分のパーティションを作成します。管理目的で OU を個別に作成した場合は、各 OU のパーティションを作成します。

複数のサーバに負荷を分散させるときには、パーティション数の制限を検討しますが、その場合もバックアップや障害回復用に最低2つのパーティションは維持します。

- ◆ 障害対策および負荷分散のために、ツリーのレプリカを最低3つ作成する。

LDAP は負荷の分散を自動では行いません。LDAP の負荷を分散させるには、第4層スイッチの使用を検討します。

- ◆ e ビジネス用のツリーを別に作成する。サーバやプリンタなど、ツリーに組み込むネットワークリソースを制限する。ユーザオブジェクトのみを格納するツリーの作成を検討する。

DirXML を使用すると、ネットワーク情報を格納する他のツリーにこのユーザツリーをリンクさせることができます。詳細については、『*DirXML Administration Guide(DirXML 管理ガイド)* (<http://www.novell.com/documentation/japanese/dirxml11a/index.html>)』を参照してください。

- ◆ 補助クラスを使用して、スキーマをカスタマイズする。

カスタマやアプリケーションで標準 `inetOrgPerson` とは異なるユーザオブジェクトが必要となる場合は、補助クラスを使用してスキーマをカスタマイズします。補助クラスを使用すると、アプリケーションの設計時にツリーを再作成しなくても、クラスで使用する属性を変更できます。

- ◆ LDIF インポートのパフォーマンスを向上させる。

Novell インポート/エクスポート変換ユーティリティを使用すると、eDirectory は処理中に各オブジェクトに索引を付けます。それによって、LDIF インポートプロセスの処理速度が遅くなる場合があります。LDIF インポートのパフォーマンスを向上させるには、作成するオブジェクトの属性からの索引付けをいったん中止し、Novell インポート/エクスポート変換ユーティリティを使用してから、属性の索引付けを再開します。

- ◆ NDS 全体で固有の共通名 (CN) を実装する。

eDirectory では、異なるコンテナ間で同じ共通名を使用できます。ただし、全体で固有の共通名を使用すれば、共通名の検索で複数の応答を処理するロジックを実装しなくて済みます。

## Novell Certificate Server について

Novell Certificate Server™ では、セキュリティコンテナオブジェクトと組織の認証局 (CA) オブジェクトを作成することによって、デジタル証明書を作成、発行、および管理できます。組織の認証局オブジェクトにより、データを保護し、安全に伝送できるようになります。NetWare Web Manager や NetWare Enterprise Web Server などの Web 関連製品には、組織の認証局オブジェクトが必要です。eDirectory サーバを初めてインストールすると、eDirectory ツリー全体のセキュリティコンテナオブジェクトと組織の認証局オブジェクトが自動的に作成され、物理的に格納されます。どちらのオブジェクトも eDirectory ツリーの最上部に作成されます。これらのオブジェクトは、移動せずに作成時の位置にそのまま保管する必要があります。

1 つの eDirectory ツリー内に存在できる組織の認証局オブジェクトは 1 つだけです。最初のサーバ上にすでに作成されている組織の認証局オブジェクトを、別のサーバに移動することはできません。組織の認証局オブジェクトを削除したり、再作成すると、その組織の認証局に関連する証明書はすべて無効になります。

**重要：**組織の認証局オブジェクトを永続的に保管しようとするサーバを、最初の eDirectory サーバにするようにします。組織の認証局オブジェクトの保管先のサーバは、ネットワーク内の継続して移動する部分であり、アクセス可能で、信頼性が高くなければなりません。

インストール中のサーバがネットワーク内の最初の eDirectory サーバでない場合、インストールプログラムは組織の認証局オブジェクトが存在する eDirectory サーバを探し、このサーバを参照します。インストールプログラムは、セキュリティコンテナにアクセスし、サーバ証明書オブジェクトを作成します。

ネットワーク内に組織の認証局オブジェクトがまったく存在しないと、Web 関連製品は機能しません。

## Novell Certificate Server でタスクを実行するのに必要な権利

Novell Certificate Server の設定に関連付けられたタスクを完了するには、管理者は、次の表に記載されているような権利を持っている必要があります。

Novell Certificate Server のタスク	必要な権利
1 台目のサーバを新しいツリーにインストールするか、まだベースセキュリティをインストールしていないツリーの 1 台目のサーバをアップグレードするためのベースセキュリティ設定	ツリーのルートのスーパーバイザ権 セキュリティコンテナのスーパーバイザ権
2 台目以降のサーバをインストールするためのベースセキュリティ設定	サーバコンテナのスーパーバイザ権 W0 オブジェクト (セキュリティコンテナ内) のスーパーバイザ権
組織の認証局の作成	セキュリティコンテナのスーパーバイザ権
サーバ認証オブジェクトの作成	サーバコンテナのスーパーバイザ権 組織の認証局オブジェクトの NDSPKI: プライベートキー属性の読み込み権

また、ルート管理者は、サブコンテナの管理者に次の権利を割り当てることによって、組織の認証局を使用する権利を委託することもできます。サブコンテナの管理者が SSL セキュリティを使用する Novell eDirectory をインストールするには、次のような権利が必要です。

- ◆ セキュリティコンテナにある組織の認証局オブジェクトの NDSPKI: プライベートキー属性の読み込み権
- ◆ KAP オブジェクト内のセキュリティコンテナにある W0 オブジェクトのスーパーバイザ権

これらの権利はグループや役割に割り当てられ、ここではすべての管理者ユーザが定義されます。Novell Certificate Server 関連の特定のタスク実行に必要な権利の詳細については、[Novell Certificate Server \(http://www.novell.com/documentation/japanese/crt252/index.html\)](http://www.novell.com/documentation/japanese/crt252/index.html) オンラインマニュアルを参照してください。

## Linux、Solaris、AIX、および HP-UX システムでの eDirectory 操作に関するセキュリティを確保する

eDirectory には、PKCS(パブリックキー暗号化サービス)が付属しています。PKCS には、PKI(Public Key Infrastructure) サービスを提供する Novell Certificate Server、NICI(Novell International Cryptographic Infrastructure)、および SAS\*-SSL サーバが含まれます。

次のセクションでは、eDirectory の操作におけるセキュリティの確保について説明します。

- ◆ 82 ページの「サーバ上に NICI がインストールおよび初期化されているかどうかを確認する」
- ◆ 82 ページの「サーバ上の NICI モジュールを初期化する」
- ◆ 83 ページの「Certificate Server(PKI サービス)を起動する」
- ◆ 83 ページの「Certificate Server(PKI サービス)を停止する」
- ◆ 83 ページの「組織の認証局オブジェクトを作成する」
- ◆ 84 ページの「サーバ認証オブジェクトを作成する」
- ◆ 84 ページの「組織の認証局の自己署名付き証明書をエクスポートする」

外部認証局の使用については、『*Novell Certificate Server Administration Guide(Novell Certificate Server 管理ガイド)* (<http://www.novell.com/documentation/japanese/crt252/index.html>)』を参照してください。

### サーバ上に NICI がインストールおよび初期化されているかどうかを確認する

次の条件を満たしているかチェックします。これらの条件は、NICI モジュールが正しくインストールおよび初期化されていることを表します。

- ◆ ファイル `/etc/nici.cfg` が存在する
- ◆ ディレクトリ `/var/novell/nici` が存在する
- ◆ ファイル `/var/novell/nici/primenici` が存在する

これらの条件を満たしていない場合、次のセクション「サーバ上の NICI モジュールを初期化する」の手順に従って操作してください。

### サーバ上の NICI モジュールを初期化する

- 1 eDirectory サーバを停止します。
  - ◆ Linux では、次のコマンドを入力します。  
`/etc/rc.d/init.d/ndsd stop`
  - ◆ Solaris では、次のコマンドを入力します。  
`/etc/init.d/ndsd stop`
  - ◆ AIX では、次のコマンドを入力します。  
`/etc/rc.d/init.d/ndsd stop`
  - ◆ HP-UX では、次のコマンドを入力します。  
`/sbin/init.d/ndsd stop`

- 2 NICI パッケージがインストールされているかどうかを確認します。
  - ◆ Linux では、次のコマンドを入力します。  
`rpm -qa | grep nici`
  - ◆ Solaris では、次のコマンドを入力します。  
`pkginfo | grep NOVLniu0`
  - ◆ AIX では、次のコマンドを入力します。  
`rpm -qa | grep nici`
  - ◆ HP-UX では、次のコマンドを入力します。  
`swlist | grep NOVLniu0`
- 3 (状況によって実行)NICI パッケージがインストールされていない場合、インストールします。  
NICI パッケージがインストールされていないと先に進みません。
- 4 パッケージに含まれている .nfc ファイルを /var/novell/nici ディレクトリにコピーします。  
/var/novell/nici/primenici プログラムを実行します。
- 5 eDirectory サーバを開始します。
  - ◆ Linux では、次のコマンドを入力します。  
`/etc/rc.d/init.d/ndsd start`
  - ◆ Solaris では、次のコマンドを入力します。  
`/etc/init.d/ndsd start`
  - ◆ AIX では、次のコマンドを入力します。  
`/etc/rc.d/init.d/ndsd start`
  - ◆ HP-UX では、次のコマンドを入力します。  
`/sbin/init.d/ndsd start`

## Certificate Server(PKI サービス) を起動する

PKI サービスを開始するには、次のコマンドを入力します。

```
npki -l
```

## Certificate Server(PKI サービス) を停止する


PKI サービスを停止するには、次のコマンドを入力します。

```
npki -u
```

## 組織の認証局オブジェクトを作成する

- 1 Novell iManager を起動します。
- 2 適切な権利を持った管理者として eDirectory ツリーにログインします。

このタスクの適切な権利を参照するには、『*Novell Certificate Server Administration Guide(Novell Certificate Server 管理ガイド)*』の「[Creating an Organizational CA \(http://www.novell.com/documentation/japanese/crt252/crtadmin/data/a2ziby.html#a2zisy5\)](http://www.novell.com/documentation/japanese/crt252/crtadmin/data/a2ziby.html#a2zisy5)」を参照してください。

- 3 [役割およびタスク] ボタン  をクリックして、[PKI Certificate Management(PKI 認証管理)] をクリックし、[認証局の作成] をクリックします。

[組織の認証局オブジェクトの作成] ウィザードが開始されます。メッセージに従ってオブジェクトを作成します。ウィザードの各ページに関する特定の情報については、[ヘルプ] をクリックします。

注: eDirectory ツリーは組織の認証局を 1 つしか格納できません。


## サーバ認証オブジェクトを作成する

サーバ認証オブジェクトは、eDirectory サーバオブジェクトを格納するコンテナに作成されます。必要に応じて、サーバ上の個々の暗号化対応アプリケーションに、別々にサーバ認証オブジェクトを作成する場合があります。または、そのサーバで使用するすべてのアプリケーションにサーバ認証オブジェクトを 1 つ作成することもあります。

注: サーバ認証オブジェクトと暗号化キーオブジェクト (KMO) は同じ意味を示しています。eDirectory オブジェクトのスキーマ名は NDSPKI: 暗号化キーです。

- 1 Novell iManager を起動します。
- 2 適切な権利を持った管理者として eDirectory ツリーにログインします。

このタスクの適切な権利を参照するには、『*Novell Certificate Server Administration Guide(Novell Certificate Server 管理ガイド)*』の「[Creating Server Certificate Objects \(http://www.novell.com/documentation/japanese/crt252/crtadmin/data/a2zibyo.html#a2zisy9\)](http://www.novell.com/documentation/japanese/crt252/crtadmin/data/a2zibyo.html#a2zisy9)」を参照してください。

- 3 [役割およびタスク] ボタン  をクリックして、[PKI Certificate Management(PKI 認証管理)] をクリックし、[サーバ証明書の作成] をクリックします。


[サーバ証明書の作成] ウィザードが開始されます。メッセージに従ってオブジェクトを作成します。ウィザードの各ページに関する特定の情報については、[ヘルプ] をクリックします。

## 組織の認証局の自己署名付き証明書をエクスポートする

自己署名付き証明書は、組織の認証局の識別情報と、組織の認証局によって署名された証明書の有効性を確認するために使用できます。

[組織の認証局] のプロパティページでは、このオブジェクトに関連付けられた証明書とプロパティを参照できます。[自己署名付き証明書] プロパティページでは、自己署名付き証明書を、暗号化対応のアプリケーションで使用するファイルにエクスポートできます。

組織の認証局に存在する自己署名付き証明書は、組織の認証局によって署名された証明書を持つサーバ認証オブジェクトのルート認証局証明書と同じものです。組織の認証局の自己署名付き証明書をルート認証局証明書として認識するサービスでは、組織の認証局によって署名された有効なユーザやサーバが許可されます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 組織の認証局オブジェクトの名前とコンテキストを指定して、[OK] をクリックします。

組織の認証局オブジェクトはセキュリティコンテナにあります。



- 4 [証明書] タブをクリックし、[Self-Signed Certificate(自己署名付き証明書)] をクリックします。
- 5 [エクスポート] をクリックします。  
証明書のエクスポートウィザードが開始されます。メッセージに従って証明書をエクスポートします。ウィザードの各ページに関する特定の情報については、[ヘルプ] をクリックします。
- 6 [証明書のエクスポートの概要] ページで、[Save the Exported Certificate to a File(エクスポートされた証明書をファイルに保存)] をクリックします。  
証明書がファイルに保存され、ルート認証局として暗号化対応アプリケーションにインポートできるようになります。
- 7 [閉じる] をクリックします。

このファイルを、eDirectory へのセキュア接続を確立するためのすべてのコマンドライン操作に組み込みます。

## ネットワーク時刻の同期

時刻同期は、ネットワーク全体で一貫したサーバ時刻を維持するためのサービスです。時刻同期は、eDirectory ではなく、サーバのオペレーティングシステムによって提供されます。eDirectory は eDirectory 自体の内部時刻に基づいて eDirectory パケットの正しい順序を維持しますが、その時刻はサーバのオペレーティングシステムから取得されます。

このセクションでは、NetWare の時刻同期と、Windows、Linux、Solaris、AIX、および HP-UX の時刻同期との統合について説明します。

### NetWare サーバで時刻を同期する

IP ネットワーク、および混合プロトコルネットワークでは、NetWare 5 サーバは IP を使用する他のサーバと時刻を同期します。NetWare 5 サーバは、timesync.nlm と NTP(ネットワーク時刻プロトコル) を使用してこの同期を行います。

NetWare 5 および 6 は、サーバで IP または IPX™ のみで使用されている場合も、両方のプロトコルが使用されている場合も、timesync.nlm を使用して時刻を同期します。timesync.nlm はサーバのインストール時にロードされます。NTP は timesync.nlm を通して設定できます。

ネットワーク内に Windows、Linux、Solaris、AIX、または HP-UX システムが含まれている場合は、時刻同期の標準である NTP を使用してサーバを同期してください。

NetWare 3 と NetWare 4 には、サードパーティの NTP 時刻サービスを使用できます。

時刻同期ソフトウェアの詳細については、[The Network Time Protocol \(http://www.ntp.org\)](http://www.ntp.org) の Web サイトを参照してください。

### NTP

NTP は、UDP プロトコル群の一部として機能します。UDP プロトコル群は、TCP/IP プロトコル群の一部として機能します。したがって、NTP を使用するコンピュータには、TCP/IP プロトコル群がロードされている必要があります。インターネットへアクセスするネットワーク内のコンピュータは、いずれもインターネット上の NTP サーバから時刻を取得できます。

NTP は、国際時刻標準である協定世界時 (UTC) と時刻を同期します。

NTP には、stratum(層) という概念が導入されています。stratum-1 サーバには、電波時計または原子時計などの正確な時計が内蔵されています。stratum-2 サーバは stratum-1 サーバから時刻を取得します。同様に、各層のサーバは 1 つ前の層のサーバから時刻を取得します。

NetWare 5 および 6 サーバでは、ntp.nlm をロードすることによって、timesync.nlm を通して NTP 時刻同期を実装できます。IP サーバで timesync.nlm によって NTP を設定すると、NTP は IP サーバと IPX サーバの両方のタイムソースとして機能するようになります。この場合、IPX サーバはセカンダリサーバとして設定する必要があります。

時刻同期の詳細については、『[Network Time Management Administration Guide\(Network Time Management 管理ガイド\)](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html) ([http://www.novell.com/documentation/japanese/nw65/time\\_enu/data/hl5k6r0y.html](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html))』および『[Network Time Protocol Administration Guide\(Network Time Protocol 管理ガイド\)](http://www.novell.com/documentation/japanese/nw65/ntp/data/aizwub2.html) (<http://www.novell.com/documentation/japanese/nw65/ntp/data/aizwub2.html>)』を参照してください。

## TIMESYNC.NLM

timesync.nlm は NetWare サーバ間の時刻を同期します。timesync.nlm は、インターネット NTP サーバなどの外部タイムソースと併せて使用できます。また、timesync.nlm が実行されているサーバに合わせて時刻を更新するよう、Novell Client™ ワークステーションを設定することもできます。

時刻同期の詳細については、『[Network Time Management Administration Guide\(Network Time Management 管理ガイド\)](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html) ([http://www.novell.com/documentation/japanese/nw65/time\\_enu/data/hl5k6r0y.html](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html))』を参照してください。

## Windows サーバで時刻を同期する

Windows NT および Windows 2000 サーバの時刻同期の詳細については、オペレーティングシステムのマニュアルを参照してください。

## Linux、Solaris、AIX、または HP-UX システムで時刻を同期する

xntpd NTP(ネットワーク時刻プロトコル) デーモンを使用すると、オペレーティングシステムデーモンは Linux、Solaris、AIX、および HP-UX サーバの時刻を同期できます。xntpd とは、インターネット標準時サーバと同期してシステム時刻を設定および保持するオペレーティングシステムデーモンです。

AIX での xntpd の実行の詳細については、『[AIX Commands Reference, Volume 6\(AIX コマンドリファレンス、第6巻\)](http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds6/xntpd.htm)』の「[xntpd Daemon](http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds6/xntpd.htm) ([http://publibn.boulder.ibm.com/doc\\_link/en\\_US/a\\_doc\\_lib/cmds/aixcmds6/xntpd.htm](http://publibn.boulder.ibm.com/doc_link/en_US/a_doc_lib/cmds/aixcmds6/xntpd.htm))」を参照してください。

Solaris での xntpd の実行の詳細については、『<http://docs.sun.com/?p=/doc/806-0625/6j9vfim2v&a=view#xntpd-1m-idx-2> (<http://docs.sun.com/?p=/doc/806-0625/6j9vfim2v&a=view#xntpd-1m-idx-2>)』を参照してください。

HP-UX での xntpd の実行の詳細については、『[Configuring NTP](http://docs.hp.com/cgi-bin/fsearch/framedisplay?top=/hpux/onlinedocs/B2355-90147/B2355-90147_top.html&con=/hpux/onlinedocs/B2355-90147/00/00/58-con.html&toc=/hpux/onlinedocs/B2355-90147/00/00/58-toc.html&searchterms=ntp%7cconfiguring&queryid=20030922-153023) ([http://docs.hp.com/cgi-bin/fsearch/framedisplay?top=/hpux/onlinedocs/B2355-90147/B2355-90147\\_top.html&con=/hpux/onlinedocs/B2355-90147/00/00/58-con.html&toc=/hpux/onlinedocs/B2355-90147/00/00/58-toc.html&searchterms=ntp%7cconfiguring&queryid=20030922-153023](http://docs.hp.com/cgi-bin/fsearch/framedisplay?top=/hpux/onlinedocs/B2355-90147/B2355-90147_top.html&con=/hpux/onlinedocs/B2355-90147/00/00/58-con.html&toc=/hpux/onlinedocs/B2355-90147/00/00/58-toc.html&searchterms=ntp%7cconfiguring&queryid=20030922-153023))』を参照してください。

Linux での ntpd の実行の詳細については、『[ntpd - Network Time Protocol \(NTP\) Daemon](http://www.eecis.udel.edu/~mills/ntp/html/ntpd.html) (<http://www.eecis.udel.edu/~mills/ntp/html/ntpd.html>)』を参照してください。

## 時刻同期を確認する

ツリー内の時刻が同期されているか確認するには、ツリーオブジェクトに対して読み書き可能以上の権利を持つ、ツリー内のいずれかのサーバから、DSRepair を実行します。

### NetWare の場合

- 1 サーバコンソールで、`dsrepair.nlm` をロードします。
- 2 [時刻同期] を選択します。

ログの解釈に関するヘルプを表示するには、<F1> を押します。

**注:** 次のコマンドを使用すると、時刻同期の問題を解決できます。

```
set timesync debug=7
```

### Windows の場合

- 1 [スタート] > [設定] > [コントロールパネル] > [Novell eDirectory Services] の順にクリックします。
- 2 `dsrepair.dlm` をクリックして、[開始] をクリックします。
- 3 [修復] をクリックし、[時刻同期] をクリックします。

### Linux、Solaris、AIX、および HP-UX

- 1 次のコマンドを実行します。

```
ndsrepair -T
```



# 3

## オブジェクトの管理

Novell® eDirectory™ には、Web ベースのネットワーク管理アプリケーションである Novell iManager 2.0.2 が含まれています。これを使用すると、eDirectory ツリーのオブジェクトを管理できます。Novell iManager の新機能および特長については、『[Novell iManager 2.0.x Administration Guide\(Novell iManager 2.0.x 管理ガイド\)](http://www.novell.com/documentation/japanese/imanager20/index.html) (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

eDirectory オブジェクトの管理には、オブジェクトの作成、変更、および操作が含まれます。たとえば、ユーザアカウントの作成や、ユーザの権利の管理などの作業が必要になることがあります。Novell iManager は次の場合に使用します。

- ◆ オブジェクトの参照、作成、編集、および編成などの基本的な管理を行う。
- ◆ ユーザのログイン名など eDirectory で使用される情報を指定して、ユーザアカウントを作成する。
- ◆ 権利の割り当て、同等セキュリティの付与、継承の阻止、および有効な権利の参照を行い、権利を管理する。詳細については、[61 ページの「権利の管理」](#)を参照してください。
- ◆ 役割ベースサービスオブジェクトを通じて特定の管理アプリケーションでの管理者の役割を定義し、役割ベースの管理を設定する。

この章では次のトピックについての情報を説明します。


- ◆ [89 ページの「オブジェクトに関連する一般的なタスク」](#)
- ◆ [94 ページの「ユーザアカウントを管理する」](#)
- ◆ [100 ページの「役割ベースサービスを設定する」](#)


### オブジェクトに関連する一般的なタスク

このセクションでは、eDirectory ツリーの管理で使用する基本的なタスクの手順について説明します。

- ◆ [90 ページの「eDirectory ツリーを参照する」](#)
- ◆ [92 ページの「オブジェクトを作成する」](#)
- ◆ [92 ページの「オブジェクトのプロパティを変更する」](#)
- ◆ [93 ページの「オブジェクトをコピーする」](#)
- ◆ [93 ページの「オブジェクトを移動する」](#)
- ◆ [93 ページの「オブジェクトを削除する」](#)
- ◆ [94 ページの「オブジェクトをリネームする」](#)

## eDirectory ツリーを参照する

Novell iManager の [オブジェクトの表示] ボタン () を使用すると、eDirectory ツリーのオブジェクトを検索したり参照したりできます。ツリーの構造を表示し、タスクを実行するオブジェクトを右クリックします。実行可能なタスクは、選択するオブジェクトの種類によって異なります。

オブジェクトの検索や参照は、Novell iManager の [eDirectory オブジェクトセクタ] ページでも行えます。Novell iManager のほとんどのエントリフィールドでは、オブジェクト名およびコンテキストを指定したり、[オブジェクトセクタ] ボタン () をクリックしてオブジェクトを検索または参照したりできます。[eDirectory オブジェクトセクタ] ページのオブジェクトを選択すると、エントリフィールドにオブジェクトおよびオブジェクトのコンテキストが挿入されます。

このセクションでは、次の情報について説明します。


- ◆ 90 ページの「[オブジェクトの表示] ボタンを使用する」
- ◆ 91 ページの「[オブジェクトセクタ] ボタンを使用する」



### [オブジェクトの表示] ボタンを使用する

次で説明する方法を使用して、管理する特定のオブジェクトを検索します。

- ◆ 90 ページの「[参照] を使用する」
- ◆ 91 ページの「[検索] を使用する」

### [参照] を使用する


- 1 Novell iManager で、[オブジェクトの表示] ボタン () をクリックします。
- 2 [参照] をクリックします。
- 3 オブジェクトを参照するには、次のオプションを使用します。

オプション	説明
	ツリー内を 1 レベル下に移動します。
	ツリー内を 1 レベル上に移動します。
コンテキスト	内容を表示するコンテナの名前を指定します。 このオプションを使用するには、コンテナの名前を指定して [適用] をクリックします。
名前	オブジェクトの名前を指定します。 このフィールドには、ワイルドカード文字としてアスタリスク (*) を使用できます。たとえば、「g*」と入力すると、Germany や Greg など g で始まるすべてのオブジェクトが検索され、「*te」と入力すると、Kate や Corporate など te で終わるすべてのエントリが検索されます。 このオプションを使用するには、名前を入力して [適用] をクリックします。

オプション	説明
タイプ	<p>検索するオブジェクトのタイプを指定します。デフォルト値は、使用可能なすべてのタイプです。</p> <p>このオプションを使用するには、ドロップダウンリストでオブジェクトのタイプを選択して、[適用] をクリックします。</p>

- 4 目的のオブジェクトを見つけたら、オブジェクトを右クリックして、実行可能なタスクのリストから選択します。

### [検索] を使用する


- 1 Novell iManager で、[オブジェクトの表示] ボタン  をクリックします。
- 2 [検索] をクリックします。
- 3 [コンテキスト] フィールドで、検索するコンテナの名前を指定します。  
[サブコンテナを検索] をクリックし、現在のコンテナにあるすべてのサブコンテナを検索対象に含めます。
- 4 [名前] フィールドで、検索するオブジェクトの名前を指定します。  
このフィールドには、ワイルドカード文字としてアスタリスク (\*) を使用できます。たとえば、「g\*」と入力すると、Germany や Greg など g で始まるすべてのオブジェクトが検索され、「\*te」と入力すると、Kate や Corporate など te で終わるすべてのエントリが検索されます。
- 5 [タイプ] ドロップダウンリストで、検索するオブジェクトのタイプを選択します。
- 6 [検索] をクリックします。
- 7 目的のオブジェクトを見つけたら、オブジェクトを右クリックして、実行可能なタスクのリストから選択します。



### [オブジェクトセレクト] ボタンを使用する

次で説明する方法を使用して、管理する特定のオブジェクトを検索します。

- ◆ 91 ページの「[参照] を使用する」
- ◆ 92 ページの「[検索] を使用する」


### [参照] を使用する

- 1 iManager のプロパティ ページで [ オブジェクト セレクト ] ボタン  をクリックします。
- 2 [参照] をクリックします。
- 3 オブジェクトを参照するには、次のオプションを使用します。



オプション	説明
	ツリー内を 1 レベル下に移動します。
	ツリー内を 1 レベル上に移動します。

オプション	説明
検索対象	内容を表示するコンテナの名前を指定して、[適用] をクリックします。
次のオブジェクトを検索	<p>オブジェクトの名前を指定します。</p> <p>このフィールドには、ワイルドカード文字としてアスタリスク (*) を使用できます。たとえば、「g*」と入力すると、Germany や Greg など g で始まるすべてのオブジェクトが検索され、「*te」と入力すると、Kate や Corporate など te で終わるすべてのエントリが検索されます。</p> <p>このオプションを使用するには、名前を入力して [適用] をクリックします。</p>


### 【検索】を使用する

- 1 iManager のプロパティ ページで [ オブジェクト セレクタ ] ボタン  をクリックします。
- 2 [検索] をクリックします。
- 3 [次から検索開始] フィールドで、検索するコンテナの名前を指定します。  
[サブコンテナを検索] をクリックし、現在のコンテナにあるすべてのサブコンテナを検索対象に含めます。
- 4 [次のオブジェクトを検索] フィールドで、検索するオブジェクトの名前を指定します。  
このフィールドには、ワイルドカード文字としてアスタリスク (\*) を使用できます。たとえば、「g\*」と入力すると、Germany や Greg など g で始まるすべてのオブジェクトが検索され、「\*te」と入力すると、Kate や Corporate など te で終わるすべてのエントリが検索されます。
- 5 [検索] をクリックします。


## オブジェクトを作成する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの作成] の順にクリックします。
- 3 使用可能なオブジェクトクラスのリストからオブジェクトを選択し、[OK] をクリックします。
- 4 必要な情報を指定して、[OK] をクリックします。  
必要な情報は、作成するオブジェクトのタイプによって異なります。詳細については、 をクリックしてください。
- 5 [OK] をクリックします。

## オブジェクトのプロパティを変更する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 変更するオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。




- 4 必要に応じてプロパティページを編集します。  
各プロパティページの詳細については、をクリックしてください。
- 5 [OK] をクリックします。

## オブジェクトをコピーする


このオプションを使用すると、既存のオブジェクトと同じ属性値を持った新しいオブジェクトを作成したり、あるオブジェクトから別のオブジェクトに属性値をコピーしたりできます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトのコピー] の順にクリックします。
- 3 [コピー元のオブジェクト] フィールドで、コピーするオブジェクトの名前およびコンテキストを指定します。
- 4 次のいずれかのオプションを選択します。
  - ◆ 新しいオブジェクトを作成して属性値をコピーする
  - ◆ 既存のオブジェクトに属性値をコピーする
- 5 アクセス制御リスト (ACL) の権利を、作成または変更するオブジェクトにコピーする場合、[ACL 権利のコピー] をクリックします。  
ACL 権利をコピーする場合、システムやネットワークの環境によってはさらに処理時間がかかる場合があります。
- 6 [OK] をクリックします。


## オブジェクトを移動する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの移動] の順にクリックします。
- 3 [オブジェクト名] フィールドで、移動するオブジェクトの名前およびコンテキストを指定します。
- 4 [移動先] フィールドで、オブジェクトを移動するコンテナを指定します。
- 5 移動する各オブジェクトについて、元の場所に別名を作成する場合は、[移動したオブジェクトの代わりに別名を作成します] を選択します。  
これにより、移動前の場所に依存するあらゆる操作は、操作を更新して移動後の場所を反映するまで、引き続き実行できます。
- 6 [OK] をクリックします。

## オブジェクトを削除する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの削除] の順にクリックします。
- 3 削除するオブジェクトの名前およびコンテキストを指定します。
- 4 [OK] をクリックします。

## オブジェクトをリネームする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトのリネーム] の順にクリックします。
- 3 [オブジェクト名] フィールドで、リネームするオブジェクトの名前およびコンテキストを指定します。
- 4 [新規オブジェクト名] フィールドで、オブジェクトの新しい名前を指定します。  
[新規オブジェクト名] フィールドには、オブジェクトのコンテキストは含めないでください。
- 5 リネームするオブジェクトに別名を作成する場合は、[リネームしたオブジェクトの代わりに別名を作成] を選択します。  
これにより、リネーム前のオブジェクト名に依存するあらゆる操作は、操作を更新してリネーム後のオブジェクト名を反映できるようになるまで、引き続き実行されます。
- 6 リネーム前のオブジェクト名を保存する場合、[古い名前を保存] をクリックします。  
これにより、リネーム前のオブジェクト名が、名前プロパティに未公認の値として追加されます。古い名前を保存しておくで、その名前に基づいてオブジェクトを検索できます。オブジェクトのリネーム後、そのオブジェクトの [General Identification(一般識別情報)] タブの [別の名前] フィールドでリネーム前の名前を参照できます。
- 7 [OK] をクリックします。

## ユーザアカウントを管理する

eDirectory のユーザアカウントの設定には、ユーザオブジェクトの作成、およびログイン制御のプロパティやユーザのネットワークコンピューティング環境の設定があります。テンプレートオブジェクトを使用すると、これらのタスクを簡単に実行できます。

ログインスクリプトを作成すると、ユーザがログインしたときに、自動的にファイルやプリンタなどの必要なネットワークリソースに接続できます。同じリソースを使用するユーザが複数いる場合、コンテナにログインスクリプトコマンドを格納して、ログインスクリプトのプロファイルを作成することができます。

このセクションでは、次の情報について説明します。

- ◆ 94 ページの「ユーザアカウントを作成および変更する」
- ◆ 95 ページの「オプションのアカウント機能を設定する」
- ◆ 97 ページの「ログインスクリプトを設定する」
- ◆ 99 ページの「リモートユーザのログイン時間制限」
- ◆ 99 ページの「ユーザアカウントを削除する」



## ユーザアカウントを作成および変更する

ユーザアカウントは、eDirectory ツリー内のユーザオブジェクトです。ユーザオブジェクトによって、ユーザのログイン名など、eDirectory で使用される情報を指定して、ネットワークリソースへのユーザのアクセスを制御できます。



このセクションでは、次の情報について説明します。

- ◆ 95 ページの「ユーザオブジェクトを作成する」
- ◆ 95 ページの「ユーザアカウントを変更する」
- ◆ 95 ページの「ユーザアカウントを有効にする」
- ◆ 95 ページの「ユーザアカウントを無効にする」


### ユーザオブジェクトを作成する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの作成] の順にクリックします。
- 3 ユーザ名とユーザの姓を指定します。
- 4 ユーザを作成するコンテナを指定します。
- 5 オプションで追加情報を指定して、[OK] をクリックします。  
使用可能なオプションの詳細については、 をクリックしてください。
- 6 [OK] をクリックします。


### ユーザアカウントを変更する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの変更] の順にクリックします。
- 3 変更するユーザの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 必要に応じてプロパティページを編集します。  
特定のプロパティの詳細については、 をクリックしてください。
- 5 [OK] をクリックします。

### ユーザアカウントを有効にする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [アカウントの有効化] の順にクリックします。
- 3 ユーザの名前とコンテキストを指定して、[OK] をクリックします。



### ユーザアカウントを無効にする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [アカウントの無効化] の順にクリックします。
- 3 ユーザの名前とコンテキストを指定して、[OK] をクリックします。



## オプションのアカウント機能を設定する

ユーザオブジェクトを作成した後、ユーザのネットワークコンピューティング環境を設定し、追加のログインセキュリティ機能を実装できます。

## ユーザのネットワークコンピューティング環境を設定する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの変更] の順にクリックします。
- 3 変更するユーザの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [全般] タブで、[使用環境] ページをクリックします。
- 5 プロパティページに入力します。  
特定のプロパティの詳細については、 をクリックしてください。
- 6 [OK] をクリックします。


## ユーザに追加のログインセキュリティを設定する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの変更] の順にクリックします。
- 3 変更するユーザの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [制限] タブで、必要なプロパティページに入力します。  
ページの詳細については、 をクリックしてください。

ページ	説明
パスワード制限	ログインパスワードを設定します。
ログイン制限	<ul style="list-style-type: none"><li>◆ アカウントを有効または無効にします。</li><li>◆ 同時ログインセッション数を制限します。</li><li>◆ ログインの有効期限およびロックアウトする日付を設定します。</li></ul>
ログイン時間制限	ユーザがログインできる時間を制限します。制限を設定し、制限時間になったときにオブジェクトがログインされていると、5 分間警告が表示され、5 分後にまだオブジェクトがログアウトされていない場合は、そのオブジェクトをログアウトします。  リモートからログインする場合は、99 ページの「リモートユーザのログイン時間制限」を参照してください。
アドレス制限	このユーザがログインするネットワークの場所 (ワークステーション) を制限します。このページで制限を設定しない場合、ユーザはネットワークのどの場所からでもログインできます。
アカウントバランス	このユーザのサーバ使用量を設定します。
不正侵入者ロックアウト	不正侵入者が検出されたためにアカウントがロックされた場合、このアカウントを操作します。不正侵入者検出の設定を管理するには、ペアレントコンテナの [不正侵入者検出] プロパティページを使用します。

- 5 [OK] をクリックします。

## コンテナ内の全ユーザの不正侵入者検出を設定する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 コンテナオブジェクトの名前とコンテキストを指定して、[OK] をクリックします。
- 4 [全般] タブで、[不正侵入者検出] ページをクリックします。
- 5 次のオプションから選択します。

オプション	説明
不正侵入者を検出する	コンテナのユーザアカウントの不正侵入者検出システムを有効にします。
不正ログイン試行回数	連続してログインに失敗して不正侵入者検出がアクティブになるまでのログイン試行回数を指定します。このコンテナ内のユーザアカウントのいずれかを使用して、連続してログインに失敗した回数がこの数値を超えた場合、不正侵入者検出がアクティブになります。この数値は、コンテナの [Login Intruder Limit(不正ログイン制限)] プロパティに格納されます。
不正ログイン回数のリセット 間隔	ここで指定する時間間隔の中で連続して失敗したログインが発生した場合、不正侵入者検出がアクティブになります。日、時間、および分を入力します。
検出後にアカウントをロック する	このコンテナ内のユーザアカウントで不正侵入者検出がアクティブになった場合に、ログインを無効にするかどうかを指定します。このチェックボックスがオンになっていない場合、不正侵入者検出がアクティブになってもアカウントはロックされません。このチェックボックスをオンにし、不正侵入者の検出によってユーザアカウントがロックされた場合、ユーザオブジェクトの [不正侵入者ロックアウト] プロパティの [ロックされたアカウント] チェックボックスをオフにして、アカウントのロックを解除できます。
日、時間、分	この3つのフィールドでは、不正侵入者検出がこのコンテナ内のユーザアカウントでアクティブになった場合にログインが無効になる時間が指定されます。指定する日、時間、および分を入力するか、デフォルトの15分を使用します。指定した時間が経過すると、ユーザアカウントのログインが再度有効になります。これらのフィールドの内容は、コンテナの [アカウントロックアウト期間] プロパティに格納されます。

- 6 [OK] をクリックします。


## ログインスクリプトを設定する

ログインスクリプトは、ユーザがログインしたときに実行される一連のコマンドです。一般的に、ユーザとファイルやプリンタなどのネットワークリソースとの接続に使用されます。ログインスクリプトは、次の順序でユーザのワークステーション上で実行されます。

1. コンテナログインスクリプト
2. プロファイルログインスクリプト
3. ユーザログインスクリプト

ログイン中に、これらのログインスクリプトで見つからないものがある場合、次のスクリプトにスキップします。何も見つからない場合、デフォルトのスクリプトが実行され、検索ドライブがユーザのデフォルトサーバ上のフォルダにマップされます。デフォルトサーバは、ユーザオブジェクトの「使用環境」プロパティページで設定されます。

## ログインスクリプトを作成する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 ログインスクリプトを作成するオブジェクトの名前とコンテキストを指定します。


ログインスクリプトの適用先	作成先
単独ユーザのみ	ユーザオブジェクト
まだ作成されていない1人以上のユーザ	テンプレートオブジェクト
1つのコンテナ内のすべてのユーザ	コンテナオブジェクト
1つ以上のコンテナの複数のユーザ	プロファイルオブジェクト

- 4 [OK] をクリックします。
- 5 [全般] タブで、[ログインスクリプト] ページを選択します。
- 6 指定するログインスクリプトコマンドを入力します。  
詳細については、『[Login Script Commands Guide\( ログインスクリプトコマンドガイド\)](http://www.novell.com/documentation/japanese/noclienu/index.html) (<http://www.novell.com/documentation/japanese/noclienu/index.html>)』を参照してください。
- 7 [OK] をクリックします。

## ユーザにプロファイルを割り当てる

プロファイルをユーザオブジェクトと関連付けることによって、ユーザのログイン中にそのプロファイルのログインスクリプトが実行されます。ユーザが、プロファイルオブジェクトのブラウザ権、およびプロファイルオブジェクトのログインスクリプトプロパティの読み込み権を持っていることを確認してください。


詳細については、65 ページの「[eDirectory オブジェクトまたはプロパティへの有効な権利を参照する](#)」を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの変更] の順にクリックします。
- 3 ログインスクリプトを作成するユーザオブジェクトの名前とコンテキストを指定します。
- 4 [OK] をクリックします。
- 5 [全般] タブで、[ログインスクリプト] ページを選択します。
- 6 プロファイルオブジェクトをこのオブジェクトに関連付けるには、[プロファイル] フィールドにプロファイルオブジェクトの名前とコンテキストを入力します。
- 7 [OK] をクリックします。

## リモートユーザのログイン時間制限

ユーザオブジェクトの [ログイン時間制限] プロパティページで、ユーザが eDirectory にログインできる時間を制限できます ( デフォルトでは、ログイン時間は制限されていません )。ログイン時間制限が設定され、制限時間になったときにユーザがログインしていると、5 分以内にログアウトするよう警告が表示されます。ユーザが 5 分経ってもまだログインしている場合、自動的にログアウトされ、保存していないデータは失われます。


ログイン要求を処理するサーバとは異なるタイムゾーンからリモートでログインする場合、ユーザに設定されたログイン時間制限の時差は調整されます。たとえば、月曜日の午前 1 時から午前 6 時までにログインが制限されており、サーバより 1 時間遅いタイムゾーンからリモートでログインする場合、このユーザの制限は午前 2 時から午前 7 時まで有効になります。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの変更] の順にクリックします。
- 3 変更するユーザの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [制限] タブで、[ログイン時間制限] をクリックします。
- 5 次のオプションから選択します。

オプション	説明
タイムグリッド	タイムグリッドのそれぞれのセルは、表示されている週の 1 日のうちの 30 分を表します。赤いセルは、このオブジェクトにログインできない制限時間を表します。灰色のセルは制限されていない時間で、このオブジェクトにログインできる時間を表します。時間制限を設定するには、指定する時間をクリックして濃い灰色にします。複数の時間を選択するには、<Shift> を押しながらセルをクリックし、該当するセルまでドラッグします。設定したログイン時間制限は、このオブジェクトの [Login Allowed Time Map( ログイン許可時間マップ )] プロパティに格納されます。
ログイン時間制限の追加	時間制限を追加するには、灰色のセルをクリックして、このオプションを選択します。
ログイン時間制限の削除	時間制限を削除するには、赤いセルをクリックして、このオプションを選択します。
更新	このボタンをクリックすると選択が有効になります。
リセット	このボタンをクリックすると、このプロパティページを開く前の状態にタイムグリッドがリセットされます。

- 6 [OK] をクリックします。

## ユーザアカウントを削除する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [ユーザ] > [ユーザの削除] の順にクリックします。
- 3 削除するユーザの名前およびコンテキストを指定します。
- 4 [OK] をクリックします。



## 役割ベースサービスを設定する

Novell iManager を使用すると、管理者は、ユーザに特定の責任を割り当てたり、これらの責任を実行するために必要なツールおよびそれに伴う権利だけを与えたりすることができます。この機能を、RBS(役割ベースサービス)といいます。

役割ベースサービスによって、タスクと呼ばれる機能、および役割と呼ばれるタスクの集まりによって決定されたオブジェクトといった特定の機能だけをユーザが使用できるようにします。ユーザが iManager にアクセスしたときの表示内容は、eDirectory 内の役割の割り当てに基づいています。表示されるのは、そのユーザに割り当てられたタスクのみです。管理するオブジェクトを探すためにツリーを参照する必要はありません。そのタスクの iManager プラグインには、タスクを実行するために必要なツールとインタフェースが用意されています。

1 人のユーザに対して複数の役割を割り当てることができます。複数のユーザに同じ役割を割り当てることができます。

役割ベースサービスは、eDirectory 内で定義されたオブジェクトとして表されます。基本の eDirectory スキーマは、iManager のインストール中に拡張されます。RBS オブジェクトタイプには、次の表のようなものがあります。

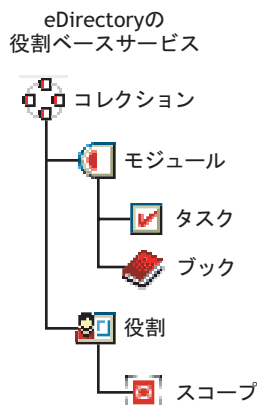
オブジェクト	説明
 RBSコレクション	<p>すべての RBS 役割オブジェクトおよびモジュールオブジェクトを格納するコンテナオブジェクト。</p> <p>RBS コレクションオブジェクトは、すべての RBS オブジェクトの最上位のコンテナです。ツリーには RBS コレクションオブジェクトをいくつでも格納できます。これらのオブジェクトには、コレクションを管理する権利を持つユーザとして「所有者」が存在します。</p> <p>RBS コレクションオブジェクトは次のコンテナのいずれかに作成できます。</p> <ul style="list-style-type: none"><li>◆ 国</li><li>◆ ドメイン</li><li>◆ 地域</li><li>◆ 組織</li><li>◆ 部門</li></ul>
 rbsRole	<p>ユーザ(メンバー)が実行を許可されたタスクを指定するコンテナオブジェクト。役割の定義には、rbsRole オブジェクトの作成や役割が実行できるタスクの指定が含まれます。</p> <p>役割メンバーには、ユーザ、グループ、組織、部門があり、ツリーの特定の範囲内の役割に関連付けられています。rbsTask オブジェクトおよび rbsBook オブジェクトは、rbsRole オブジェクトに割り当てられます。</p> <p>rbsRole オブジェクトは、RBS コレクションコンテナ内でのみ作成できます。</p>



オブジェクト	説明
 rbsModule	<p>rbsTask オブジェクトおよび rbsBook オブジェクトを格納するコンテナオブジェクトです。rbsModule オブジェクトには、タスクやブックを定義する製品の名前 (たとえば、eDirectory Maintenance Utilities、NMA Management、Novell Certificate Server Access など) を表すモジュール名の属性があります。</p> <p>rbsModule オブジェクトは、RBS コレクションコンテナ内でのみ作成できます。</p>
 rbsTask	<p>ログインパスワードのリセットなど、特定の機能を表すリーフオブジェクトです。</p> <p>rbsTask オブジェクトは、rbsModule コンテナ内でのみ格納されます。</p>
 rbsBook	<p>ブックに割り当てられる一連のページを含むリーフオブジェクトです。rbsBook は、1 つ以上の役割および 1 つ以上のオブジェクトクラスタイプに割り当てることができます。</p> <p>rbsBook オブジェクトは、rbsModule コンテナ内でのみ格納されます。</p>
 rbsScope	<p>ユーザオブジェクトごとに割り当てずに ACL 割り当てを実行するためのリーフオブジェクトです。rbsScope オブジェクトは、役割が実行されるツリー内のコンテキストを表し、rbsRole オブジェクトに関連付けられます。このオブジェクトはグループクラスから権利を継承します。ユーザオブジェクトは rbsScope オブジェクトに割り当てられます。これらのオブジェクトは、関連付けられるツリーのスコープを参照します。</p> <p>このオブジェクトは、必要な場合は動的に作成され、不要になれば自動的に削除されます。rbsScope オブジェクトは rbsRole コンテナ内でのみ格納されます。</p> <p><b>警告</b> : Scope オブジェクトの環境設定は変更しないでください。設定を変更することによって深刻な問題が発生し、システムが故障する可能性があります。</p>

RBS オブジェクトは、次の図に示されているように eDirectory ツリーに属しています。

図 22 eDirectory ツリー内の RBS オブジェクト



## RBS 役割を定義する

RBS 役割は、ユーザが実行を許可されるタスクを指定します。RBS 役割の定義には、rbsRole オブジェクトの作成や、役割が実行できるタスク、およびユーザ、グループ、またはこれらのタスクを実行できるコンテナオブジェクトの指定などがあります。Novell iManager プラグイン (製品パッケージ) には、変更可能な定義済み RBS 役割が提供されている場合もあります。


RBS 役割が実行できるタスクは、eDirectory ツリー内では rbsTask オブジェクトとして公開されます。これらのオブジェクトは、製品パッケージのインストールの際に自動的に追加されます。オブジェクトは 1 つ以上の rbsModule に編成され、異なる機能を持つ製品モジュールに対応するコンテナとなります。

役割へのメンバーの割り当てについての詳細は、[103 ページの「RBS 役割のメンバーシップおよびスコープを割り当てる」](#)を参照してください。

- ◆ [102 ページの「役割オブジェクトを作成する」](#)
- ◆ [102 ページの「役割に関連付けられたタスクを変更する」](#)
- ◆ [103 ページの「RBS 役割のメンバーシップおよびスコープを割り当てる」](#)
- ◆ [103 ページの「役割ベースサービスオブジェクトを削除する」](#)

### 役割オブジェクトを作成する



Create iManager Role ウィザードを使用して、新しい rbsRole オブジェクトを作成します。新しい rbsRole オブジェクトを作成する場合は、他の rbsRole オブジェクトが属している同じ RBS コレクションコンテナ (たとえば、役割ベースサービスコレクションコンテナ) 内に作成することをお勧めします。

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [役割の設定] > [iManager の役割を作成] の順にクリックします。
- 3 Create iManager Role ウィザードの手順に従って操作します。

役割へのメンバーの追加についての詳細は、[103 ページの「カスタム RBS タスクを定義する」](#)を参照してください。

### 役割に関連付けられたタスクを変更する

各 RBS 役割には、使用可能なタスクが関連付けられます。特定の役割に割り当てるタスクは、必要に応じてタスクを追加したり削除したりすることで選択できます。

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [役割の設定] > [iManager の役割を変更] の順にクリックします。
- 3 役割のタスクを追加または削除するには、変更する役割の左にある [タスクの変更] ボタン  をクリックします。
- 4 [割り当てられたタスク] リストでタスクを追加または削除します。
- 5 [OK] をクリックします。

## RBS 役割のメンバーシップおよびスコープを割り当てる



所属する組織に必要な RBS 役割を定義すると、それぞれの役割にメンバーを割り当てるができます。その際、それぞれのメンバーが役割の機能を使用できるスコープを指定します。スコープは、この役割を実行できる eDirectory ツリー内の場所またはコンテキストです。

役割へのユーザの割り当ては、次の方法で行うことができます。


- ◆ 直接
- ◆ グループおよび動的グループの割り当てによる方法。役割に割り当てられているグループまたは動的グループのメンバーであれば、ユーザはその役割にアクセスできます。
- ◆ 職種割り当てによる方法。役割に割り当てられている職種に所属する場合は、ユーザはその役割にアクセスできます。
- ◆ コンテナ割り当てによる方法。ユーザオブジェクトは、そのペアレントコンテナが割り当てられたすべての役割にアクセスできます。さらにツリーのルートまで遡るコンテナの役割にもアクセスできます。

役割との関連付けは、さまざまなスコープで何度も実行できます。また、同じタスクを複数のメンバーに割り当てることもできます。

役割のメンバーシップおよびスコープを割り当てるには、次の操作を実行します。

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [役割の設定] > [iManager の役割を変更] の順にクリックします。
- 3 役割のメンバーを追加または削除するには、変更する役割の左にある [メンバーの変更] ボタン  をクリックします。
- 4 [名前] フィールドには、オブジェクトの名前 (ユーザ、グループ、またはコンテナオブジェクト) およびコンテキストを指定します。
- 5 [スコープ] フィールドには、組織または部門オブジェクトの名前およびコンテキストを指定します。
- 6 [追加] をクリックし、[OK] をクリックします。


## 役割ベースサービスオブジェクトを削除する

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [役割の設定] > [役割の削除] の順にクリックします。
- 3 削除する RBS 役割の名前およびコンテキストを指定します。
- 4 [OK] をクリックします。

## カスタム RBS タスクを定義する


- ◆ [104 ページの「iManager タスクを作成する」](#)
- ◆ [104 ページの「サーバ管理タスクを作成する」](#)
- ◆ [104 ページの「役割の割り当てを変更する」](#)
- ◆ [104 ページの「タスクを削除する」](#)

## iManager タスクを作成する


- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [タスクの設定] > [iManager のタスクを作成] の順にクリックします。
- 3 Task Builder の手順に従ってカスタムタスクを作成します。

## サーバ管理タスクを作成する


[サーバ管理タスクの作成] ウィザードを使用してカスタムタスクを作成し、サーバのサービスにアクセスします。システム管理者は、サービスがサーバ上で使用可能かどうかを確認する必要があります。

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [タスクの設定] > [サーバ管理タスクの作成] の順にクリックします。
- 3 Create Server Administration Task ウィザードの手順に従って操作します。

## 役割の割り当てを変更する

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [タスクの設定] > [役割の割り当てを変更] の順にクリックします。
- 3 変更するタスクの名前およびコンテキストを指定して、[次へ] をクリックします。
- 4 割り当てを変更する役割を [使用可能な役割] カラムから [割り当て役割] カラムへ移動します。
- 5 [OK] をクリックします。

## タスクを削除する

- 1 Novell iManager で、[設定] ボタン  をクリックします。
- 2 [タスクの設定] > [タスクの削除] の順にクリックします。
- 3 削除するタスクの名前およびコンテキストを指定して、[OK] をクリックします。

# 4

## スキーマの管理

Novell® eDirectory™ ツリーのスキーマには、このツリーに含めることができるオブジェクト（ユーザ、グループ、プリンタなど）のクラスを定義します。スキーマによって、各オブジェクトタイプを構成する属性（プロパティ）が指定されます。属性には、オブジェクトの作成に不可欠な必須属性と、必要に応じて指定できるオプション属性があります。

**eDirectory** オブジェクトはそれぞれオブジェクトクラスに属し、オブジェクトクラスはオブジェクトに関連付けることのできる属性を指定します。すべての属性は一連の属性タイプに基づくもので、属性タイプもまた、一連の標準的な属性構文に基づいています。

**eDirectory** スキーマは、各オブジェクトの構造を制御するだけでなく、eDirectory ツリー内でのオブジェクト間の関係も制御します。スキーマルールを設定すると、オブジェクトは他の従属オブジェクトを含むことができます。このように、スキーマによって eDirectory ツリーの構造が決まります。

組織が必要とする情報の変化に応じて、スキーマに変更を加える必要が出てくる場合があります。たとえば、ユーザオブジェクトに、以前は Fax 番号が不要であっても、現在は必要であるとしてします。この場合、Fax 番号を必須属性とした新しいユーザクラスを作成し、ユーザオブジェクトの作成に、この新しいユーザクラスを使用できます。

Novell iManager でスキーマ管理の役割を持つ場合、ツリーのスーパーバイザ権を持つユーザは、そのツリーのスキーマをカスタマイズして次のようなタスクを実行できます。

- ◆ スキーマ内のすべてのクラスおよび属性の一覧を表示する。
- ◆ 既存のスキーマにクラスまたは属性を追加して、スキーマを拡張する。
- ◆ クラスを作成する。名前を付けてから、属性、フラグ、追加先コンテナ、および属性の継承元のペアレントクラスを指定することにより行います。
- ◆ 名前を付けてから、構文およびフラグを設定して、属性を作成する。
- ◆ 既存クラスへ属性を追加する。
- ◆ 使用されていない、あるいは古くなったクラスまたは属性を削除する。
- ◆ 潜在的な問題を発見および解決する。

この章では次のトピックについての情報を説明します。

- ◆ [106 ページの「スキーマの拡張」](#)
- ◆ [109 ページの「スキーマの表示」](#)
- ◆ [110 ページの「手動でスキーマを拡張する」](#)
- ◆ [112 ページの「eDirectory 8.7 に追加されたスキーマフラグ」](#)
- ◆ [114 ページの「eMBox クライアントを使用してスキーマ操作を実行する」](#)

スキーマ情報の詳細については、『[NDS Schema Reference\(NDS スキーマリファレンス\)](http://developer.novell.com/ndk/doc/ndslib/index.html?schem_enu/data/h4q1mn1i.html) ([http://developer.novell.com/ndk/doc/ndslib/index.html?schem\\_enu/data/h4q1mn1i.html](http://developer.novell.com/ndk/doc/ndslib/index.html?schem_enu/data/h4q1mn1i.html))』を参照してください。

# スキーマの拡張

新しいクラスや属性を作成することにより、ツリーのスキーマを拡張できます。eDirectory ツリーのスキーマを拡張するには、ツリー全体に対するスーパーバイザ権が必要です。

次の作業により、スキーマを拡張できます。


- ◆ クラスを作成する
- ◆ クラスを削除する
- ◆ 属性を作成する
- ◆ クラスへオプション属性を追加する
- ◆ 属性を削除する

次の作業により、補助属性のスキーマを拡張できます。

- ◆ 補助クラスを作成する
- ◆ 補助クラスのプロパティでオブジェクトを拡張する
- ◆ オブジェクトの補助プロパティを変更する
- ◆ オブジェクトから補助プロパティを削除する

## クラスを作成する

組織の必要条件の変化に応じて、既存のスキーマに対しクラスを追加できます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [クラスの作成] の順にクリックします。
- 3 [クラスの作成] ウィザードの指示に従って、オブジェクトクラスを定義します。

ウィザードの各段階で、[ヘルプ] が利用できます。

オブジェクトクラスに追加するカスタムプロパティを定義する場合は、ウィザードを終了し、最初にカスタムプロパティを定義します。詳細については、[107 ページの「属性を作成する」](#)を参照してください。


## クラスを削除する

使用されていないクラスは、そのクラスが eDirectory ツリーのベーススキーマの一部でない限り、削除できます。iManager では、ローカルにレプリカ作成されたパーティションで現在使用されているクラスだけは削除できません。

次のような場合に、スキーマからクラスを削除できます。


- ◆ 2つのツリーをマージし、クラスの違いを解決した場合
- ◆ 特定のクラスが不要になった場合

クラスを削除するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [クラスの削除] の順にクリックします。
- 3 削除するクラスを選択します。  
削除可能なクラスのみが表示されます。
- 4 [削除] をクリックします。

## 属性を作成する

独自のカスタムタイプ属性を定義し、これを既存のオブジェクトクラスのオプション属性に追加できます。ただし、既存のクラスに必須属性を追加することはできません。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [属性の作成] の順にクリックします。
- 3 [属性の作成] ウィザードの指示に従って、新しい属性を定義します。  
ウィザードの各段階で、[ヘルプ] が利用できます。




## クラスへオプション属性を追加する

既存のクラスにオプションの属性を追加できます。これは、次のような場合に必要になります。

- ◆ 組織の必要とする情報が変化した場合
- ◆ ツリーのマージを準備している場合

**注:** 必須属性は、クラスの作成時にのみ定義できます。

オプション属性クラスを追加するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [属性の追加] の順にクリックします。
- 3 属性を追加するクラスを選択して、[OK] をクリックします。
- 4 使用可能なオプション属性リストで、追加する属性を選択して  をクリックし、追加するオプション属性リストにこれらの属性を追加します。  
誤って属性を追加したり、後で属性を削除したい場合は、追加するオプション属性リストで属性を選択し、 をクリックして追加する属性のリストから削除します。
- 5 [OK] をクリックします。

このクラスにオブジェクトを作成すると、ここで追加したプロパティを含むオブジェクトが作成されます。追加したプロパティの値を設定するには、オブジェクトの [その他] 一般プロパティページを使用します。

**ヒント:** 既存のクラスは、このページにある Current Attributes(現在の属性) リストに追加して変更できます。削除できるのは、追加してからまだ [OK] をクリックしていない属性のみです。前に追加した属性や保存した属性は削除できません。


## 属性を削除する

使用されていない属性は、その属性が eDirectory ツリーのベーススキーマの一部でない限り、削除できます。

次のような場合に、スキーマから属性を削除できます。

- ◆ 2つのツリーをマージし、属性の違いを解決した場合
- ◆ 特定の属性が不要になった場合

属性を削除するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [属性の削除] の順にクリックします。


- 3 削除する属性を選択します。  
削除可能な属性のみが表示されます。
- 4 [削除] をクリックします。

## 補助クラスを作成する


補助クラスとは、あるオブジェクトクラス全体ではなく、特定の eDirectory オブジェクトインスタンスに追加される一連のプロパティ ( 属性 ) です。たとえば、電子メールアプリケーション用として、eDirectory ツリーのスキーマに電子メールプロパティ補助クラスを追加し、必要に応じて、このプロパティを個別のオブジェクトに拡張できます。

スキーママネージャを使用すると、独自の補助クラスを定義できます。補助クラスで定義したプロパティを使用して、個別のオブジェクトを拡張できます。

補助クラスを作成するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [クラスの作成] の順にクリックします。
- 3 クラス名およびオプションで ASN1 ID を指定して、[次へ] をクリックします。
- 4 クラスフラグを設定する場合は [補助クラス] をクリックし、[次へ] をクリックします。
- 5 [クラスの作成] ウィザードの指示に従って、新しい補助クラスを定義します。  
ウィザードの各段階で、[ヘルプ] が利用できます。


## 補助クラスのプロパティでオブジェクトを拡張する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [オブジェクトの拡張] の順にクリックします。
- 3 拡張するオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 使用する補助クラスが [現在の補助クラスの拡張] の下にすでに表示されているかどうかに応じて、適切な操作を実行します。

補助クラスがリストに 表示されている	操作
はい	この手順を終了します。代わりに実行する手順については、 <a href="#">108 ページの「オブジェクトの補助プロパティを変更する」</a> を参照してください。
いいえ	[追加] をクリックし、補助クラスを選択して、[OK] をクリックします。

- 5 [閉じる] をクリックします。


## オブジェクトの補助プロパティを変更する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。



- 3 変更するオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [全般] タブで、[その他] ページをクリックします。
- 5 表示された画面で、必要な属性値を設定します。
  - ◆ 値のない属性をダブルクリックし、値のある属性のリストに追加します。
  - ◆ 値のある属性を選択し、[編集] をクリックして属性を編集するか、[削除] をクリックして属性を削除します。
  - ◆ 正しく設定するためには、各プロパティの構文について理解している必要があります。詳細については、『[Understanding Schema Manager\(スキーママネージャについて\)](http://www.novell.com/documentation/japanese/nds8/docui/index.html#./usnds/schm_enu/data/hnpkthb2.html) ([http://www.novell.com/documentation/japanese/nds8/docui/index.html#./usnds/schm\\_enu/data/hnpkthb2.html](http://www.novell.com/documentation/japanese/nds8/docui/index.html#./usnds/schm_enu/data/hnpkthb2.html))』を参照してください。
- 6 [適用] をクリックし、[OK] をクリックします。

## オブジェクトから補助プロパティを削除する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [オブジェクトの拡張] の順にクリックします。
- 3 拡張するオブジェクトの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [現在の補助クラスの拡張] のリストから、削除するプロパティが定義されている補助クラスを選択します。
- 5 [削除] をクリックし、[OK] をクリックします。

これにより、オブジェクトに最初から定義されていたプロパティを除き、補助クラスによって追加されたすべてのプロパティが削除されます。
- 6 [閉じる] をクリックします。


## スキーマの表示


スキーマが組織の情報のニーズに合ったものかどうかを評価するために、スキーマを表示したり印刷することができます。組織が大きく、また複雑になると、スキーマをカスタマイズする必要も大きくなります。しかし、小規模な組織でも、特別な記録を必要とする場合があるかもしれません。このような場合、スキーマの表示や印刷は、ベーススキーマにどのような拡張が必要かを定めるのに役立ちます。

## クラス情報を参照する



iManager の [クラス情報] ページには、選択されたクラスに関する情報が表示され、そこで属性を追加できます。このページに表示されているほとんどの情報は、クラスが作成されたときに指定されたものです。オプション属性の中には、後で追加されたものもあります。

クラスの作成中に、そのクラスを他のクラスの属性を継承するように指定した場合は、継承された属性はペアレントクラスの中に分類されます。たとえば、オブジェクトクラスがペアレントクラスで必須属性である場合、このページには、選択されたクラスの必須属性として表示されます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。

- 2 [スキーマ] > [クラス情報] の順にクリックします。
- 3 情報を表示するクラスを選択し、[表示] をクリックします。  
詳細については、「」を参照してください。

## 属性情報を表示する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [スキーマ] > [属性の情報] の順にクリックします。
- 3 情報を表示する属性を選択し、[表示] をクリックします。  
詳細については、「」を参照してください。

## 手動でスキーマを拡張する

.sch 拡張子の付いたファイルを使用して、手動で eDirectory スキーマを拡張できます。  
このセクションでは、次の情報について説明します。

- ◆ 110 ページの「[NetWare でスキーマを拡張する](#)」
- ◆ 110 ページの「[Windows でスキーマを拡張する](#)」
- ◆ 111 ページの「[Linux、Solaris、AIX、または HP-UX システムでスキーマを拡張する](#)」

## NetWare でスキーマを拡張する

NWConfig.nlm を使用して、NetWare サーバのスキーマを拡張します。eDirectory に付属しているスキーマファイル (\*.sch) は、sys:\system\schema ディレクトリにインストールされます。

- 1 サーバコンソールで、「nwconfig」と入力します。
- 2 [ディレクトリオプション] > [スキーマの拡張] の順に選択します。
- 3 管理権を持つユーザとしてログインします。
- 4 <F3> を選択して異なるパスを指定し、sys:\system\schema(または \*.sch ファイルのパス) とスキーマファイルの名前を入力します。
- 5 <Enter> キーを押します。

## Windows でスキーマを拡張する

NDSCons.exe を使用して、Windows サーバのスキーマを拡張します。eDirectory に付属しているスキーマファイル (\*.sch) は、デフォルトで C:\Novell\NDS ディレクトリにインストールされます。

- 1 [スタート] > [設定] > [コントロールパネル] > [Novell eDirectory Services] の順にクリックします。
- 2 install.dlm をクリックし、[開始] をクリックします。
- 3 [追加のスキーマファイルのインストール] をクリックし、[次へ] をクリックします。
- 4 管理権を持つユーザとしてログインし、[OK] をクリックします。
- 5 スキーマファイルのパスと名前を指定します。
- 6 [完了] をクリックします。

## Linux、Solaris、AIX、または HP-UX システムでスキーマを拡張する

次のセクションでは、Linux、Solaris、AIX、および HP-UX システムでのスキーマの拡張について説明します。

- ◆ 111 ページの「[ndssch ユーティリティを使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張する](#)」
- ◆ 111 ページの「[RFC 2307 スキーマを拡張する](#)」

### ndssch ユーティリティを使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張する

Novell iManager のほかにも、eDirectory スキーマ拡張ユーティリティ `ndssch` を使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張することができます。ツリーのスキーマの変更処理では、スキーマファイル (.sch) に指定された属性とクラスが使用されます。.sch ファイルで指定した内容に従って、属性とクラスの関連付けが作成されます。

- 1 次の構文を使用します。

```
ndssch [-h hostname[:port]] [-t tree_name] admin-FDN schemafilename...
ndssch [-h hostname[:port]] [-t tree_name] [-d admin_FDN schemafilename
[description]]...
```

ndssch のパラメータ	説明
<code>-h hostname</code>	スキーマを拡張するサーバの名前または IP アドレス。指定したサーバが属しているツリーのスキーマが拡張されます。スキーマを拡張するホスト上にツリーがある場合のみ、オプションで指定するパラメータです。それ以外の場合は、必須パラメータです。
<code>port</code>	サーバポート。
<code>-t tree_name</code>	スキーマを拡張するツリーの名前。このパラメータの指定は任意です。/etc/nds.conf ファイルに指定された値がデフォルトのツリー名として使用されます。詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「 <a href="#">環境設定パラメータ</a> 」を参照してください。
<code>admin-FDN</code>	ツリーに対する eDirectory 管理権を持つユーザのフルコンテキスト付きの名前。
<code>schemafilename</code>	拡張するスキーマについての情報が入力されたファイルの名前。
<code>-d, description</code>	このオプションが使用されている場合、各スキーマファイルはスキーマファイルの説明を伴っています。

### RFC 2307 スキーマを拡張する

RFC 2307 (<http://www.ietf.org/rfc/rfc2307.txt>) に定義されている属性とオブジェクトクラスは、ユーザまたはグループ関連、および NIS 関連のものです。ユーザまたはグループ関連の定義は、`/usr/lib/nds-modules/schema/rfc2307-usergroup.sch` ファイルにコンパイルされます。NIS 関連の定義は、`/usr/lib/nds-modules/schema/rfc2307-nis.sch` ファイルにコンパイルされます。それぞれに対応する LDIF 形式のファイルもあります (ユーザ/グループ関連は `/usr/lib/nds-modules/schema/rfc2307-usergroup.ldif`、NIS 関連は `/usr/lib/nds-modules/schema/rfc2307-nis.ldif`)。

RFC 2307 スキーマを拡張するには、`ndssch` ユーティリティまたは `ldapmodify` ツールを使用します。

- ◆ 112 ページの「[ndssch ユーティリティを使用する](#)」
- ◆ 112 ページの「[ldapmodify ユーティリティを使用する](#)」

### ndssch ユーティリティを使用する

次のいずれかのコマンドを入力します。

```
ndssch -t /usr/lib/nds-schema/rfc2307-usergroup.sch
```

または

```
ndssch -t /usr/lib/nds-schema/rfc2307-nis.sch
```

パラメータ	説明
-t	スキーマを拡張するツリーの名前。このパラメータの指定は任意です。このパラメータが指定されていない場合、 <code>/etc/nds.conf</code> ファイルに指定されたツリー名が使用されます。

### ldapmodify ユーティリティを使用する

次のいずれかのコマンドを入力します。

```
ldapmodify -h -D -w -f /usr/lib/nds-schema/rfc2307-usergroup.ldif
```

または

```
ldapmodify -h -D -w -f /usr/lib/nds-schema/rfc2307-nis.ldif
```

パラメータ	説明
-h <i>ldaphost</i>	LDAP サーバの実行場所となっている代替ホストを指定します。
-D <i>binddn</i>	<i>binddn</i> を使用して X.500 ディレクトリにバインドします。binddn には、RFC 1779 に定義されている文字列表現の DN を指定します。
-w <i>passwd</i>	簡易認証のパスワードとして、 <i>passwd</i> を使用します。
-f <i>file</i>	エントリ情報を標準入力ではなく <i>file</i> から読み出します。

## eDirectory 8.7 に追加されたスキーマフラグ

eDirectory 8.7 には、`READ_FILTERED` および `BOTH_MANAGED` スキーマフラグが追加されました。

`READ_FILTERED` は、属性が LDAP オペレーショナル属性であることを示すために使用されます。LDAP がこのフラグを使用するのは、スキーマを読み込んで属性が「オペレーショナル」であることを知らせる必要がある場合です。内部定義のスキーマ属性には、このフラグが設定されたものもあります。LDAP 「オペレーショナル」定義には、3つのスキーマフラグがあります。新しい `READ_FILTERED` フラグとは別に、既存のフラグとして「オペレーショナル」を示すものには `READ_ONLY` フラグと `HIDDEN` フラグがあります。これらのフラグのいずれかがスキーマ定義に存在する場合、LDAP は属性を「オペレーショナル」として扱い、特に必要がない場合はその属性を返しません。

BOTH\_MANAGED は、セキュリティ権利を強制する新しいメカニズムです。これは識別名構文の属性にのみ重要なフラグです。この属性に設定された場合、要求している接続には、ターゲットオブジェクトと属性、およびターゲット属性によって参照されているオブジェクトの両方に対する権利が必要になります。これは、現在の WRITE\_MANAGED フラグの機能を拡張したものです。このフラグは、現在ベーススキーマ属性には設定されていません。この新しいセキュリティ動作は eDirectory 8.7.x サーバ上でのみ実行されるため、このフラグに関連する動作に矛盾がないようにするには、ツリー全体を eDirectory 8.7 以降に更新する必要があります。

eDirectory 8.7.x サーバだけがこれらの新しいフラグを認識するため、ルートパーティションのコピーを格納する eDirectory 8.7.x サーバによってのみ、これらのフラグをスキーマ定義に設定できます。これは、ルートを格納するサーバのみがスキーマを変更できるためです。通常インストールの新規サーバや、ルートパーティションを保持していない既存サーバをアップグレードしたサーバでは、これらの新しいフラグをツリー内のスキーマに追加することはできません。

これらの新しい機能のいずれかをツリーで有効にする場合は、スキーマが正常に拡張されていて、これらの新しいフラグを追加できることを確認する必要があります。確認する方法には次の 2 つがあります。1 つは、ルートパーティションの書き込み可能なコピーを持つサーバを選択して、eDirectory 8.7 以降にアップグレードすることです。これにより、自動的にスキーマが正しく拡張され、新しいフラグに対応します。

2 つ目の方法はさらに込み入ったものです。次の手順に従って操作します。

- 1 新しく 8.7.x サーバをインストールするか、ツリー内の既存のサーバをアップグレードします。このサーバが [Root] のコピーを保持している必要はありません。
- 2 ルートパーティションのコピーをこの新しいサーバに手動で追加します。
- 3 次に示す適切なスキーマ拡張ファイルをこのサーバ上で再実行し、スキーマを拡張します。

プラットフォーム	手順
Windows	install.dlm をロードし、[追加のスキーマファイルのインストール] をクリックします。
NetWare	nwconfig をロードし、[ディレクトリオプション] > [スキーマの拡張] の順にクリックします。
Linux、Solaris、AIX、HP-UX	ndsschユーティリティを使用します。詳細については、 <a href="#">111 ページの「ndssch ユーティリティを使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張する」</a> を参照してください。

- 4 これらの新しいフラグが設定された新しいスキーマファイルをインストールします。
- 5 (オプション)スキーマの同期後は、このサーバからルートレプリカを削除できます。

注：これらの新しいスキーマフラグにより、オプションの機能が有効になります。新しい機能を必要としない場合は、スキーマ定義にこれらの新しいフラグが存在しなくても、ツリー内での eDirectory の通常操作に問題が起こることはありません。READ\_FILTERED フラグの場合は、属性の定義によっては存在しないことがあります。このため、オブジェクトのすべての属性に対する LDAP 読み込み要求によって、フラグが存在すれば読み込まないはずのデータが余分に取得される場合があります。READ\_FILTERED フラグを含む属性の中には、READ\_ONLY フラグまたは HIDDEN フラグが存在するために、やはりオペレーショナルとして扱われるものもあります。BOTH\_MANAGED フラグは、すべてのサーバがアップグレードされたツリーでのみ有効になります。その環境の中でのみ、この機能を矛盾なく操作することが可能なためです。

## eMBox クライアントを使用してスキーマ操作を実行する

eMBox (eDirectory Management Toolbox) クライアントはコマンドライン Java クライアントで、これを使用すると DSSchema 操作にリモートでアクセスできます。DSSchema eMTool を使用すると、スキーマの同期、リモートスキーマのインポート、新しいスキーマエポックの宣言、ローカルスキーマのリセット、グローバルスキーマの更新などを実行できます (通常、DSRepair を使用して実行する操作です。詳細については、[225 ページの「スキーマの保守」](#)を参照してください)。

eDirectory の一部としてサーバ上に `emboxclient.jar` ファイルがインストールされます。JVM を実行しているコンピュータならどこからでも実行できます。eMBox クライアントの詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

### DSSchema eMTool を使用する

- 1 非対話式モードで eMBox クライアントを実行するには、コマンドラインに次のコマンドを入力します。

```
java -cp path_to_the_file/emboxclient.jar embox -i
```

(クラスパスにすでに `emboxclient.jar` ファイルが含まれている場合、`java embox -i` と入力するだけです)。

次の eMBox クライアントのプロンプトが表示されます。

```
eMBox Client>
```

- 2 修復するサーバにログインするには、次のように入力します。

```
login -sserver_name_or_IP_address -pport_number  
-username.context -wpassword -n
```

ポート番号は通常 80 または 8008 です。ただし、すでにそのポートを使用している Web サーバが存在する場合は異なります。-n オプションを使用すると、セキュリティ保護されていない接続を開始します。

eMBox クライアントはログインが成功したかどうかを表示します。

- 3 次の構文を使用して修復コマンドを入力します。

```
dsschema.task options
```

次に例を示します。

`dsschema.rst` は、このサーバのスキーマを同期するようツリーのルートのマスタレプリカに要求します。

`dsschema.irs -nMyTree` によって、MyTree というツリーからリモートスキーマがインポートされます。

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。

eMBox クライアントは修復が成功したかどうかを表示します。

DSSchema eMTool オプションの詳細については、[115 ページの「DSSchema eMTool オプション」](#)を参照してください。

- 4 eMBox クライアントからログアウトするには、次のコマンドを入力します。

```
logout
```

- 5 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

## DSSchema eMTool オプション

次の表に、DSSchema eMToolオプションを示します。eMBoxクライアントでlist-tdsschemaコマンドを使用して、DSSchema オプションの詳細を表示することもできます。詳細については、[483 ページの「eMTool とそのサービスを表示する」](#)を参照してください。

オプション	説明
rst	ツリーのルートのマスタレプリカのスキーマをこのサーバに同期します。
irs -ntree_name	別のツリーからリモートスキーマをインポートします。
dse	ルートのマスタレプリカを持つサーバ上で新しいスキーマエポックを宣言します。
rls	ローカルスキーマを、ルートパーティションのマスタレプリカがあるサーバからのコピーでリセットします。
gsu	Post NetWare 5 レベルへのグローバルスキーマ更新を実行します。
scc	ドメインクラスのスキーマサーキュラ包含規則を追加します。





# 5

## パーティションおよびレプリカの管理

パーティションは、eDirectory ツリー内の個別のデータユニットを構成する Novell® eDirectory™ データベースの論理区分です。システム管理者は、パーティションを利用して eDirectory 情報を格納し、レプリカを作成します。各パーティションは、コンテナオブジェクト、コンテナオブジェクトに含まれるすべてのオブジェクト、およびこれらのオブジェクトについての情報で構成されます。パーティションには、ファイルシステムに関する情報、またはパーティションに含まれるディレクトリやファイルに関する情報は含まれません。

各サーバに eDirectory データベース全体のコピーを保存する代わりに、eDirectory パーティションのコピーを作成してそれをネットワーク内の複数のサーバ上で保存できます。パーティションのそれぞれのコピーはレプリカと呼ばれます。各 eDirectory パーティションのレプリカは任意の数だけ作成することができ、任意のサーバに保存できます。レプリカのタイプには、マスタ、読み書き可能、読み込み専用、サブオーディネートリファレンス、フィルタ済み読み書き可能、およびフィルタ済み読み込み専用があります。

次の表で、レプリカタイプについて説明します。

レプリカ	説明
マスタ、読み書き可能、および読み込み専用	特定のパーティションのすべてのオブジェクトおよび属性が含まれます。
サブオーディネートリファレンス	ツリーの接続のために使用されます。
フィルタ済みレプリカ	<p>パーティション全体の情報のサブセットで、必要なクラスおよび属性が含まれます。必要なクラスおよび属性は、サーバのレプリケーションフィルタによって定義されず、レプリケーションフィルタは、インバウンド同期やローカルでの変更時にレプリカに含めることのできるクラスおよび属性を識別するために使用されます。</p> <p>フィルタ済みレプリカによって、管理者はまばらで断片的なレプリカを作成できます。</p> <ul style="list-style-type: none"><li>指定したオブジェクトクラスだけが含まれるスパーレプリカ</li><li>指定した属性だけが含まれる断片レプリカ</li></ul> <p>フィルタ済みレプリカの機能によって、アプリケーションが eDirectory に格納されているデータを取得するときのレスポンスが迅速になります。また、フィルタ済みレプリカを使用すると、1つのサーバにより多くのレプリカを格納できます。</p>

レプリカ	説明
読み書き可能フィルタ済みレプリカ	サーバのレプリケーションフィルタのサブセットであるクラスおよび属性をローカルで変更できます。ただし、これらのレプリカを作成できるのは、レプリケーションフィルタ内にそのクラスの必須属性がすべて含まれている場合のみです。
読み込み専用フィルタ済みレプリカ	ローカルで変更できません。

この章では、パーティションおよびレプリカの管理方法を説明します。

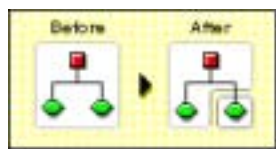
- ◆ 118 ページの「パーティションの作成」
- ◆ 119 ページの「パーティションのマージ」
- ◆ 120 ページの「パーティションの移動」
- ◆ 121 ページの「パーティションの作成操作またはマージ操作のキャンセル」
- ◆ 121 ページの「レプリカの管理」
- ◆ 125 ページの「フィルタ済みレプリカを設定し管理する」
- ◆ 127 ページの「パーティションおよびレプリカを表示する」

## パーティションの作成

パーティションを作成すると、ツリーの論理区分が作成されます。これらの論理区分は、ネットワーク内にある別の eDirectory サーバ間で複製したり配布することができます。

新しいパーティションを作成すると、ペアレントパーティションが分割されて 2 つのパーティションになります。新しいパーティションは、次の図で示されるように、チャイルドパーティションになります。

図 23 パーティションの分割前と分割後




たとえば、1 つの部門を選択し、これを新しいパーティションとして作成すると、選択した部門およびその従属オブジェクトすべてがペアレントパーティションから分割されます。

選択した部門は、新しいパーティションのルートになります。新しいパーティションのレプリカは、ペアレントパーティションのレプリカと同じサーバに存在します。また、新しいパーティションのオブジェクトは、そのパーティションのルートオブジェクトに属します。

レプリカすべてを新しいパーティション情報と同期する必要があるため、パーティションの作成には時間がかかる場合があります。パーティションの作成中に別のパーティション操作を実行しようとする、パーティションが使用中であることを示すメッセージが表示されます。

新しいパーティションのレプリカリストを参照し、リスト内のレプリカがすべてオンの状態であれば、操作が完了していることがわかります。状態は自動的にリフレッシュされないため、画面を定期的に手動でリフレッシュします。

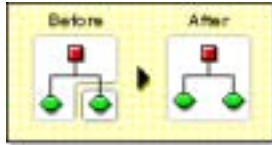
パーティションを作成するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタンをクリックします。
- 2 [パーティションとレプリカ] > [パーティションの作成] の順にクリックします。
- 3 新しいパーティションを作成するコンテナの名前およびコンテキストを指定して、[OK] をクリックします。

## パーティションのマージ

パーティションをそのペアレントパーティションにマージすると、選択したパーティションおよびそのレプリカがペアレントパーティションに結合されます。パーティションは削除できません。次の図に示されるように、パーティションはマージおよび作成だけを行い、ディレクトリツリーがどのように論理区分に分割されるかを定義します。

図 24 パーティションのマージ前とマージ後



パーティションとそのペアレントパーティションをマージする理由として、次のようなことが考えられます。

- ◆ 2つのパーティションのディレクトリ情報が密接に関連している。
- ◆ 従属パーティションを削除する場合に、その中のオブジェクトを残したい。
- ◆ パーティションのオブジェクトを削除する。
- ◆ パーティションのすべてのレプリカを削除する(パーティションのマスタレプリカを削除する唯一の方法は、パーティションをそのペアレントにマージすることです)。
- ◆ コンテナを移動した後で ( ルートパーティションに従属パーティションがない場合のみ)、このコンテナをパーティションとする必要がなくなった。
- ◆ 会社の組織に変更が生じたため、パーティション構造を変更することでディレクトリツリーを再設計する。

パーティションが大きくなる ( 何百ものオブジェクトが含まれる ) とネットワークの応答時間が遅くなるため、パーティションを分割することを検討します。

ツリーのルートパーティションは最上位のパーティションであり、マージするペアレントパーティションがないため、マージできません。


サーバで処理が完了すると、パーティションがマージされます。パーティションのサイズ、ネットワークトラフィック、サーバの環境設定などによって異なりますが、この操作の完了にはかなり時間がかかる場合があります。

**重要:** パーティションのマージを行う前に、両パーティションの同期を点検し、続行する前にすべてのエラーを修正します。エラーを修正することにより、ディレクトリでの問題を切り離し、エラーの伝播や新しいエラーの発生を防ぐことができます。

パーティションのマージを行う前に、マージするパーティションのレプリカ (サブオーディネートリファレンスを含む) を持つサーバすべてが稼働していることを確認します。サーバが停止中の場合、eDirectory はサーバのレプリカを読み込むことができず、操作を完了できません。

パーティションのマージの処理中にエラーが表示された場合は、そのつどエラーを解決します。操作を続行してエラーを修正しないでください。さらにエラーが発生するおそれがあります。

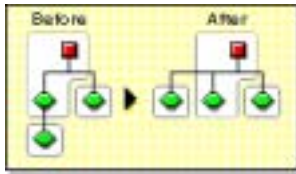
チャイルドパーティションをペアレントパーティションとマージするには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [パーティションのマージ] の順にクリックします。
- 3 ペアレントパーティションとマージするパーティションの名前およびコンテキストを指定して、[OK] をクリックします。

## パーティションの移動

パーティションの移動により、ディレクトリツリー内のサブツリーを移動できます。ルートパーティションオブジェクト (コンテナオブジェクト) に従属パーティションがない場合にのみ、このオブジェクトを移動できます。

図 25 パーティションの移動前と移動後



パーティションを移動する場合、eDirectory 包含規則に従ってください。たとえば、部門は現在のツリーのルートの直下には移動できません。これは、ルートの包含規則により移動が許されているのが、地域、国、または組織であるためです。

パーティションを移動すると、eDirectory は、ルートパーティションオブジェクトへの参照すべてを変更します。オブジェクトの共通名は変更されませんが、コンテナ (およびその従属コンテナすべて) の完全識別名は変更されます。

パーティションを移動するときに、オプションを選択して、移動するコンテナの代わりに別名オブジェクトを作成することもできます。これにより、ユーザは引き続きネットワークにログインでき、元のディレクトリ位置にオブジェクトを見つけることができます。

作成した別名オブジェクトは、移動したコンテナと同じ共通名を持ち、そのコンテナの新しい完全識別名を参照します。

**重要:** パーティションを移動したときに、移動したパーティションの代わりとなる別名オブジェクトを作成しないと、パーティションの新しい位置を知らないユーザは元のディレクトリ位置でオブジェクトを見つけようとし、ディレクトリツリーにあるパーティションオブジェクトを見つけることが困難になります。

また、ワークステーションの NAME CONTEXT パラメータがディレクトリツリーコンテナのオリジナルの位置に設定されている場合、これにより、クライアントワークステーションがログインできないという問題が起きるおそれがあります。


オブジェクトを移動すると、オブジェクトの名前コンテキストが変更されるため、移動したオブジェクトを参照する名前コンテキストのユーザは、NAME CONTEXT パラメータを更新する必要があります。これにより、名前コンテキストは、オブジェクトの新しい名前を参照するようになります。

コンテナオブジェクトの移動の後に、ユーザの NAME CONTEXT パラメータを自動的に更新するには、NCUPDATE ユーティリティを使用します。

移動したパーティションをパーティションとして使用しない場合は、それをペアレントパーティションとマージします。

パーティションを移動する前に、ディレクトリツリーが正しく同期していることを確認します。移動元パーティションと移動先パーティションのどちらかで同期のエラーが生じた場合は、パーティションの移動操作を実行しないでください。実行する前に、同期エラーを解決します。

パーティションを移動するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [パーティションの移動] の順にクリックします。
- 3 [オブジェクト名] フィールドに移動するパーティションオブジェクトの名前およびコンテキストを指定します。
- 4 [移動先] フィールドに移動するコンテナの名前およびコンテキストを指定します。
- 5 移動するパーティションについて、元の場所に別名を作成する場合は、[移動したオブジェクトの代わりに別名を作成します] を選択します。  
これにより、移動前の場所に依存するあらゆる操作は、操作を更新して移動後の場所を反映するまで、引き続き実行できます。
- 6 [OK] をクリックします。

## パーティションの作成操作またはマージ操作のキャンセル

変更を確定する段階まで操作が到達していない場合、パーティションの作成またはマージをキャンセルできます。この機能を使用して、操作を終了できます。また、eDirectory ネットワークが、eDirectory エラーを返したり、パーティション操作に続く同期に失敗した場合にもこの機能を使用します。

ディレクトリツリーのレプリカで同期エラーが生じていると、操作を中止しても問題が解決しない場合があります。ただし、初期のトラブルシューティングオプションとして、この機能を使用できます。

サーバが停止している、またはその他の理由でサーバが使用できない場合、パーティション操作を完了できるようにサーバがネットワークから見えるようにするか、操作を中止します。データベースが壊れているために eDirectory による同期ができない場合、実行中のパーティション操作すべてを中止する必要があります。

含まれるレプリカの数、サーバの可視性、および既存のワイヤトランフィックの量により、パーティション操作でネットワーク間の完全な同期を行うには、かなりの時間がかかります。

パーティションが使用中であることを示すエラーが表示されても、操作を中止する必要はありません。パーティションのサイズ、接続性の問題などによって異なりますが、通常、パーティション操作は 24 時間以内に完了します。この時間内で操作が完了しない場合には、実行中の操作を中止します。

## レプリカの管理


レプリカの追加、削除、またはレプリカタイプの変更をする前に、ターゲットレプリカの位置を慎重に計画します。76 ページの「ツリーのレプリカ作成に関するガイドライン」を参照してください。





## レプリカを追加する

次の機能をディレクトリに提供するために、レプリカをサーバに追加します。

- ◆ 障害対策
- ◆ データへのより高速なアクセス
- ◆ WAN リンク上でのより高速なアクセス
- ◆ 設定コンテキスト中のオブジェクトへのアクセス (バインダリサービスを使用)

レプリカを追加するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 レプリカを作成するパーティションまたはサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [レプリカの追加] をクリックします。
- 5 パーティションまたはサーバの名前およびコンテキストを指定します。
- 6 次のレプリカタイプのいずれかを選択します。

レプリカタイプ	説明
 読み書き可能	ユーザは、新しいレプリカの内容の読み込みと変更を両方行うことができます。このパーティションの eDirectory オブジェクトを管理するユーザの近くに変更可能なレプリカがない場合は、このオプションを選択します。
 読み込み専用	ユーザは、新しいレプリカの内容を読み込むことはできませんが、変更はできません。このパーティションの eDirectory オブジェクトを読み込むだけで、変更を行わないユーザの近くにレプリカがない場合は、このオプションを選択します。
 フィルタ済み読み書き可能	ユーザは、新しいレプリカの内容の読み込みと変更を両方行うことができますが、このレプリカの内容は、eDirectory オブジェクトおよびフィルタで指定されたプロパティのタイプに制限されます。
 フィルタ済み読み込み専用	ユーザは、新しいレプリカの内容を読み込んでも変更はできず、このレプリカの内容は、eDirectory オブジェクトおよびフィルタで指定されたプロパティのタイプに制限されます。

- 7 [OK] をクリックします。

詳細については、[49 ページの「レプリカのタイプ」](#)を参照してください。

## レプリカを削除する

レプリカを削除すると、パーティションのレプリカはサーバから削除されます。

サーバをディレクトリツリーから削除する場合は、その前に、レプリカをサーバから削除します。レプリカを削除することで、サーバを削除するときに起きる問題を減らすことができます。

また、レプリカの削除により、ネットワークの同期トラフィックの量も削減できます。通常、パーティションに7個以上のレプリカは必要ありません。

マスタレプリカまたはサブオーディネートリファレンスは、削除できません。

マスタレプリカを削除するには、次の2つのオプションを利用します。

- ◆ マスタレプリカをパーティションの別のレプリカを含むサーバへ移動し、そのレプリカを新しいマスタレプリカにします。

これにより、元のマスタレプリカは自動的に読み書き可能レプリカに変更され、削除できるようになります。

- ◆ パーティションをそのペアレントパーティションとマージします。

これにより、パーティションのレプリカとペアレントパーティションのレプリカがマージされ、そのサーバからレプリカが削除されます。マージによりパーティションの境界は削除されますが、オブジェクトは削除されません。オブジェクトは、「結合された」パーティションのレプリカを持つ各サーバに残ります。



レプリカを削除する場合、次の点に注意します。

- ◆ 障害対策として、異なるサーバ上に各パーティションのレプリカを3つ以上保持します。
- ◆ レプリカを削除すると、ターゲットサーバ上のディレクトリデータベースのコピーが削除されます。

データベースは、ネットワークの別のサーバで引き続きアクセスできます。また、レプリカが含まれていたサーバも、eDirectory で引き続き機能します。

サブオーディネートリファレンスレプリカは、削除や管理ができません。サーバがパーティションのレプリカを含む場合、サブオーディネートリファレンスレプリカは、eDirectory により自動的にサーバ上に作成されます。ただし、サーバがパーティションのチャイルドレプリカを含む場合には、作成されません。

レプリカを削除するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 削除するレプリカを格納するパーティションまたはサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 削除するレプリカの左にある  をクリックします。
- 5 [OK] をクリックします。

## レプリカタイプを変更する

レプリカタイプを変更して、レプリカ情報へのアクセスを制御します。たとえば、既存の読み書き可能レプリカを読み込み専用レプリカに変更して、ユーザがレプリカに書き込んだり、ディレクトリデータを変更できないようにします。


読み書き可能レプリカ、または読み込み専用レプリカのタイプを変更できます。マスタレプリカのタイプは変更できませんが、読み書き可能レプリカまたは読み込み専用レプリカは、マスタレプリカに変更できます。これにより、元のマスタレプリカは、自動的に読み書き可能レプリカに変更されます。






通常、ほとんどのレプリカは、読み書き可能レプリカとして使用されます。読み書き可能レプリカには、クライアント操作による書き込みができます。変更が加えられると、

読み書き可能レプリカは同期情報を送信します。読み込み専用レプリカには、クライアント操作による書き込みができません。しかし、レプリカを同期すると、読み込み専用レプリカは更新されます。

サブオーディネートリファレンスのレプリカタイプは、変更できません。サブオーディネートリファレンスがあるサーバに、パーティションのレプリカを配置するには、レプリカの追加操作を行う必要があります。サブオーディネートリファレンスレプリカは、パーティションの完全なコピーではありません。サブオーディネートリファレンスレプリカの配置および管理は、eDirectory で制御します。サーバがパーティションのレプリカを含む場合、サブオーディネートリファレンスレプリカは、eDirectory により自動的にサーバ上に作成されます。ただし、サーバがパーティションのチャイルドレプリカを含む場合には、作成されません。

レプリカタイプを変更するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 変更するレプリカを格納するパーティションまたはサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 変更するレプリカのレプリカタイプ ([タイプ] カラム内) をクリックします。
- 5 新しいレプリカタイプをクリックし、[OK] をクリックします。

レプリカタイプ	説明
 マスタ	ユーザは、このレプリカの内容の読み込みと変更を行うことができ、このレプリカは、下位パーティションの作成やマージなど、このパーティションに影響を与える将来のパーティション処理の出発点となります。マスタレプリカは、パーティションごとに1つだけ設定できます。
 読み書き可能	ユーザは新しいレプリカの内容の読み込みと変更を行うことができます。このパーティションの eDirectory オブジェクトを管理するユーザの近くに変更可能なレプリカがない場合は、このオプションを選択します。
 読み込み専用	ユーザは新しいレプリカの内容を読み込むことができますが、変更はできません。このパーティションの eDirectory オブジェクトを読み込むだけで、変更を行わないユーザの近くにレプリカがない場合は、このオプションを選択します。
 フィルタ済み読み書き可能	ユーザは新しいレプリカの内容の読み込みと変更を行うことはできますが、内容はフィルタで指定された eDirectory オブジェクトとプロパティのタイプに制限されます。
 フィルタ済み読み込み専用	ユーザは新しいレプリカの内容を読み込んでも変更はできませんが、内容はフィルタで指定された eDirectory オブジェクトとプロパティのタイプに制限されます。

- 6 [OK] をクリックします。

詳細については、[49 ページの「レプリカのタイプ」](#)を参照してください。



## フィルタ済みレプリカを設定し管理する

フィルタ済みレプリカには、eDirectory パーティションの情報のフィルタ済みサブセット (オブジェクトまたはオブジェクトクラス、およびこれらのオブジェクトの属性と値のフィルタ済みセット) が保存されます。

管理者は、フィルタ済みレプリカのセットを保持する eDirectory サーバを作成するためにフィルタ済みレプリカ機能を使用します。フィルタ済みレプリカのセットには、同期するオブジェクトおよび属性のみが含まれます。


このため、iManager では、フィルタ済みレプリカのパーティションスコープおよびフィルタを作成できるツールが用意されています。スコープとは、単に、サーバ上でレプリカを保存するパーティションのセットです。一方、レプリケーションフィルタには、サーバのフィルタ済みレプリカセットに含める eDirectory クラスおよび属性のセットが定義されます。この結果、eDirectory サーバには、ツリー内の多くのパーティションから必要なデータを抽出し明確に定義されたデータセットが保存されることとなります。

サーバのパーティションスコープおよびレプリケーションフィルタの記述は、eDirectory に格納され、iManager のサーバオブジェクトまたは [パーティションとレプリカ] 役割によって管理できます。

- [125 ページの「フィルタ処理済レプリカウィザードを使用する」](#)
- [126 ページの「パーティションスコープを定義する」](#)
- [126 ページの「サーバフィルタを設定する」](#)

## フィルタ処理済レプリカウィザードを使用する

フィルタ処理済レプリカウィザードを使用すると、サーバのレプリケーションフィルタおよびパーティションスコープを、表示される手順に従って簡単に設定できます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [フィルタ処理済レプリカウィザード] の順にクリックします。
- 3 フィルタ済みレプリカを設定するサーバを指定し、[次へ] をクリックします。
- 4 選択されたサーバに設定されたフィルタのクラスおよび属性を定義するには、[フィルタセットの定義] をクリックします。

レプリケーションフィルタには、サーバのフィルタ済みレプリカセットに保存したい eDirectory クラスおよび属性のセットが含まれます。フィルタセットの定義の詳細については、[126 ページの「サーバフィルタを設定する」](#)を参照してください。

- 5 [次へ] をクリックします。
- 6 このサーバのパーティションスコープを定義するには、[パーティションスコープの定義] をクリックします。

パーティションスコープの詳細については、[126 ページの「パーティションスコープを定義する」](#)を参照してください。


- 7 [次へ] > [完了] の順にクリックします。

## パーティションスコープを定義する


パーティションスコープは、サーバ上でレプリカを保存するパーティションのセットです。iManager の [レプリカビュー] ページには、eDirectory ツリーのパーティションの階層が表示されます。個別のパーティション、指定した分岐のパーティションセット、またはツリー内のすべてのパーティションを選択できます。次に、サーバに追加するこれらのパーティションのレプリカタイプを選択するか、既存のレプリカタイプを変更します。

サーバには、完全なレプリカもフィルタ済みレプリカも保存できます。詳細については、[51 ページの「フィルタ済みレプリカ」](#)を参照してください。


### eDirectory サーバのレプリカを表示する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 表示するサーバの名前およびコンテキストを指定して [OK] をクリックし、このサーバのレプリカリストを表示します。

### eDirectory サーバにフィルタ済みレプリカを追加する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 フィルタ済みレプリカを追加するサーバの名前およびコンテキストを指定し、[OK] をクリックします。
- 4 [レプリカの追加] をクリックします。
- 5 パーティションの名前およびコンテキストを指定します。
- 6 [フィルタ済み読み書き可能] または [フィルタ済み読み込み専用] をクリックし、[OK] をクリックします。

### 完全なレプリカをフィルタ済みレプリカへ変更する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 変更するレプリカを格納するパーティションまたはサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 変更するレプリカのレプリカタイプ ([タイプ] カラム内) をクリックします。
- 5 [フィルタ済み読み書き可能] または [フィルタ済み読み込み専用] をクリックし、[OK] をクリックします。

## サーバフィルタを設定する


サーバレプリケーションフィルタには、サーバのフィルタ済みレプリカセットに保存したい eDirectory クラスおよび属性のセットが含まれます。どのサーバオブジェクトからでもフィルタを設定できます。フィルタ済みレプリカの場合、サーバごとにフィルタを1つだけ作成できます。つまり、ある eDirectory サーバ用に定義されているフィルタは、そのサーバ上のすべてのフィルタ済みレプリカに適用されます。ただし、完全なレプリカにはフィルタは適用されません。

サーバのフィルタは必要に応じて変更できますが、変更するとレプリカの再同期が発生するため時間がかかります。サーバの機能については、慎重に計画することを推奨します。


次の方法のいずれかで、サーバのフィルタを設定または変更できます。

- ◆ [127 ページの「レプリカビューを使用する」](#)
- ◆ [127 ページの「サーバオブジェクトを使用する」](#)

## レプリカビューを使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 変更するレプリカを格納するパーティションまたはサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 変更するサーバまたはパーティションの [Edit in the Filter( フィルタ内の編集 )] カラムをクリックします。
- 5 適切なクラスおよび属性を追加し、[OK] をクリックします。
- 6 [完了] をクリックします。

## サーバオブジェクトを使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 変更するレプリカを格納するサーバの名前およびコンテキストを指定して、[OK] をクリックします。
- 4 [レプリカ] タブをクリックします。
- 5 このサーバにフィルタが定義されていなかった場合、[フィルタは空です] をクリックして [フィルタの編集] ダイアログボックスを開き、目的のクラスおよび属性を追加します。  
または  
[Copy Filter From( フィルタからコピー )] をクリックし、コピーするフィルタを持つオブジェクト ( 別のサーバなど ) を参照します。
- 6 既存のフィルタを編集するには、フィルタ内のハイパーリンク付きアイテムをクリックし、[フィルタの編集] ダイアログボックスを開いて、目的のクラスおよび属性を追加または削除します。


## パーティションおよびレプリカを表示する

このセクションでは、次の情報について説明します。

- ◆ [128 ページの「サーバのパーティションを表示する」](#)
- ◆ [128 ページの「パーティションレプリカを表示する」](#)
- ◆ [128 ページの「パーティションに関する情報を表示する」](#)
- ◆ [128 ページの「パーティションの階層を表示する」](#)
- ◆ [128 ページの「レプリカに関する情報を表示する」](#)

## サーバのパーティションを表示する

Novell iManager では、サーバに割り当てられたパーティションを表示できます。サーバオブジェクトをディレクトリツリーから削除しようとする場合、サーバに格納されているパーティションの表示が必要となる場合があります。この場合、オブジェクトを削除する前に、削除するレプリカを表示できます。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 サーバオブジェクトの名前とコンテキストを入力して、[OK] をクリックします。

## パーティションレプリカを表示する

この操作により、次を識別できます。


- ◆ パーティションのレプリカが存在するサーバ
- ◆ パーティションのマスタレプリカのホストとなっているサーバ
- ◆ 読み書き可能レプリカ、読み込み専用レプリカ、およびサブオーディネートリファレンスレプリカを含むサーバ
- ◆ 各パーティションレプリカの状態

パーティションのレプリカを表示するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 パーティションの名前とコンテキストを入力して、[OK] をクリックします。


## パーティションに関する情報を表示する

パーティションに関する情報 ( 成功した最新の同期や最近試みた同期など ) を表示する主な目的は、パーティションの同期情報を確認することです

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [パーティションの情報を表示します] の順にクリックします。
- 3 パーティションの名前とコンテキストを入力して、[OK] をクリックします。

## パーティションの階層を表示する

iManager では、パーティションの階層を容易に表示できます。コンテナオブジェクトを展開して、ペアレントパーティションやチャイルドパーティションを表示できます。


パーティションのルートであるコンテナには、次のアイコンが表示されます。 

## レプリカに関する情報を表示する

レプリカに関する情報を表示する主な目的は、レプリカの状態を確認することです。eDirectory レプリカの状態は、それが実行しているパーティションや複製の操作によってさまざまです。次の表では、iManager で表示されるレプリカの状態を説明しています。

状態	説明
オン	現在パーティションや複製の操作を実行していない
新規	サーバに新しいレプリカとして追加中
停止中	サーバから削除中
停止	サーバからの削除が完了
マスタ開始	マスタレプリカへ変更中
マスタ完了	マスタレプリカへの変更が完了
タイプの変更	他のレプリカタイプへの変更中
ロック状態	パーティションの移動または修復の操作の準備が滞っている
移動へ移行	パーティションの移動操作を開始中
移動	パーティションの移動操作中
分割へ移行	パーティションの分割操作 (チャイルドパーティションの作成) を開始中
分割	パーティションの分割 (チャイルドパーティションの作成) 操作中
結合	ペアレントパーティションへのマージ中
オンへ移行	オン状態へ戻る直前
不明	iManager で認識できない状態

レプリカの情報を表示するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。
- 3 パーティションまたはサーバの名前およびコンテキストを入力して、[OK] をクリックします。



# 6

## Novell eDirectory 管理ユーティリティ

この章では、Novell<sup>®</sup> eDirectory<sup>™</sup> で提供されている次のユーティリティについて説明します。

- ◆ 131 ページの「Novell インポート / エクスポート変換ユーティリティ」
- ◆ 163 ページの「インデックスマネージャ」
- ◆ 168 ページの「プレディケートデータ」
- ◆ 168 ページの「eDirectory Service Manager」

### Novell インポート / エクスポート変換ユーティリティ

Novell インポート / エクスポート変換ユーティリティは次の操作に使用できます。

- ◆ LDIF ファイルから LDAP ディレクトリへのデータのインポート
- ◆ LDAP ディレクトリから LDIF ファイルへのデータのエクスポート
- ◆ LDAP サーバ間でのデータの移行
- ◆ スキーマの比較と更新の実行
- ◆ テンプレートを使用した eDirectory への情報のロード
- ◆ SCH ファイルから LDAP ディレクトリへのスキーマのインポート

Novell インポート / エクスポート変換ユーティリティは、形式に応じてデータを読み書きするための一連のハンドラを管理します。データを読み込むハンドラをソースハンドラと呼び、データを書き込むハンドラをターゲットハンドラと呼びます。1 つの実行可能モジュールがソースハンドラとターゲットハンドラの両方として機能することもあります。Novell インポート / エクスポート変換エンジンは、ソースハンドラからデータを受け取ってそれを処理し、処理したデータをターゲットハンドラに渡します。

たとえば、LDIF データを LDAP ディレクトリにインポートする場合、Novell インポート / エクスポート変換エンジンは、LDIF ソースハンドラを使用して LDIF ファイルを読み込み、読み込んだデータを LDAP ターゲットハンドラを使用して LDAP ディレクトリサーバに送信します。LDIF ファイルの構文、構造、およびデバッグの詳細については、[498 ページの「LDIF ファイルのトラブルシューティング」](#)を参照してください。

Novell インポート / エクスポート変換クライアントユーティリティは、コマンドライン、ConsoleOne<sup>®</sup> スナップイン、または Novell iManager のインポート / エクスポート変換ウィザードから実行できます。ただし、コマンド区切りのデータに対応したハンドラは、コマンドラインユーティリティと Novell iManager でのみ使用できます。

Novell インポート / エクスポート変換ユーティリティは、次のどちらの方法でも使用できます。

- ◆ 132 ページの「Novell iManager インポート / エクスポート変換ウィザードを使用する」
- ◆ 135 ページの「コマンドラインインタフェースを使用する」

Novell インポート / エクスポート変換エンジンには、ウィザードからもコマンドラインインタフェースからもアクセスできます。ただし、ソースハンドラとターゲットハンドラの組み合わせの選択肢は、コマンドラインインタフェースを使用する場合のほうが多くなります。

NDS および eDirectory の旧バージョンで提供されていた BULKLOAD ユーティリティと ZONEIMPORT ユーティリティは、どちらも Novell インポート / エクスポート変換ユーティリティに変更されました。


## Novell iManager インポート / エクスポート変換ウィザードを使用する

インポート / エクスポート変換ウィザードは次の操作に使用できます。

- ◆ LDIF、区切りテキストファイル、スキーマファイル、または LOAD ファイルからのデータのインポート。
- ◆ LDIF ファイルへのデータのエクスポート。
- ◆ サーバ間でのデータの移行。

Novell iManager の使用方法とアクセス方法の詳細については、『[Novell iManager 2.0.x Administration Guide \(Novell iManager 2.0.x 管理ガイド\)](http://www.novell.com/documentation/japanese/imanager20/index.html) (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

### データをファイルからインポートする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルからデータをインポート] をクリックし、[次へ] をクリックします。
- 4 インポートするファイルのタイプを選択します。
- 5 インポートするデータを含むファイルの名前と適切なオプションを指定して、[次へ] をクリックします。

このページのオプションは、選択したファイルのタイプによって異なります。使用可能なオプションの詳細については、[ヘルプ] をクリックしてください。


- 6 データをインポートする LDAP サーバを指定します。
- 7 次の表に示すように、適切なオプションを追加します。

オプション	説明
サーバのDNS名/IPアドレス	相手 LDAP サーバの DNS 名または IP アドレス
ポート	相手 LDAP サーバのポート番号 ( 整数 )
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法 )
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

- 8 [次へ] > [完了] の順にクリックします。



## データをファイルへエクスポートする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート 変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルにデータをエクスポート] > [次へ] の順にクリックします。
- 4 エクスポートするエントリが格納されている LDAP サーバを指定します。  
[詳細設定] を使用して、LDAP ソースハンドラの追加オプションを設定します。使用可能なオプションの詳細については、[ヘルプ] をクリックしてください。
- 5 次の表に示すように、適切なオプションを追加します。


オプション	説明
サーバの DNS 名 / IP アドレス	ソース LDAP サーバの DNS 名または IP アドレス
ポート	ソース LDAP サーバのポート番号 ( 整数 )
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法 )
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

- 6 [次へ] をクリックします。
- 7 エクスポートするエントリの検索条件を次のように指定します。

オプション	説明
ベース DN	検索リクエストのベース識別名 このフィールドを指定しなかった場合、デフォルトのベース DN である ""( 空の文字列 ) が使用されます。
スコープ	検索リクエストのスコープ
フィルタ	RFC 1558 準拠の検索フィルタ デフォルトは「objectclass=*」です。
属性	検索エントリごとに取得する属性

- 8 [次へ] をクリックします。
- 9 エクスポートするファイルのタイプを選択します。  
エクスポートされたファイルは、一時的な場所に保存されます。このファイルは、インポート / エクスポート 変換ウィザードの最後でダウンロードできます。
- 10 [次へ] > [完了] の順にクリックします。

## LDAP サーバ間でデータを移行する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート 変換ウィザード] の順にクリックします。
- 3 [サーバ間でデータを移行] > [次へ] の順にクリックします。
- 4 移行するエントリが格納されている LDAP サーバを指定します。

[詳細設定] を使用して、LDAP ソースハンドラの追加オプションを設定します。使用可能なオプションの詳細については、[ヘルプ] をクリックしてください。

- 5 次の表に示すように、適切なオプションを追加します。

オプション	説明
サーバの DNS 名 / IP アドレス	ソース LDAP サーバの DNS 名または IP アドレス
ポート	ソース LDAP サーバのポート番号 ( 整数 )
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法 )
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

- 6 [次へ] をクリックします。
- 7 移行するエントリの検索条件を次のように指定します。

オプション	説明
ベース DN	検索リクエストのベース識別名  このフィールドを指定しなかった場合、デフォルトのベース DN である ""( 空の文字列 ) が使用されます。
スコープ	検索リクエストのスコープ
フィルタ	RFC 2254 準拠の検索フィルタ  デフォルトは「objectclass=*」です。
属性	検索エントリごとに取得する属性

- 8 [次へ] をクリックします。
- 9 データを移行する LDAP サーバを指定します。
- 10 [次へ] > [完了] の順にクリックします。

注: スキーマが各 LDAP サービスで整合性を保っていることを確認します。

## コマンドラインインタフェースを使用する

Novell インポート / エクスポート変換ユーティリティのコマンドラインバージョンは、次の操作に使用できます。

- ◆ LDIF のインポート
- ◆ LDIF のエクスポート
- ◆ コンマ区切りデータのインポート
- ◆ コンマ区切りデータのエクスポート
- ◆ LDAP サーバ間でのデータの移行
- ◆ スキーマの比較と更新
- ◆ テンプレートを使用した eDirectory への情報のロード
- ◆ スキーマのインポート

Novell インポート / エクスポート変換ウィザードは、Novell iManager の一部としてインストールされます。Win32\* バージョン (`ice.exe`) と NetWare® バージョン (`ice.nlm`) の両方がインストールされます。Linux、Solaris、AIX、および HP-UX の各システムの場合、インポート / エクスポートユーティリティは、NOVLice パッケージに含まれています。

### Novell インポート / エクスポート変換構文

Novell インポート / エクスポート変換ユーティリティは、次の構文で起動します。

```
ice general_options  
-S[LDIF | LDAP | DELIM | LOAD | SCH] source_options  
-D[LDIF | LDAP | DELIM] destination_options
```

またはスキーマキャッシュを使用する場合は、次の構文になります。

```
ice -C schema_options  
-S[LDIF | LDAP] source_options  
-D[LDIF | LDAP] destination_options
```

スキーマキャッシュを使用して更新を実行する場合、LDIF ファイルはターゲットとして有効ではありません。

一般オプションの指定は任意です。ただし、指定する場合はソースハンドラオプションやターゲットハンドラオプションより前に指定します。`-S`(ソース)ハンドラセクションと `-D`(ターゲット)ハンドラセクションはどちらを先に指定してもかまいません。

利用できるソースハンドラとターゲットハンドラは次のとおりです。

- ◆ [138 ページの「LDIF ソースハンドラのオプション」](#)
- ◆ [138 ページの「LDIF ターゲットハンドラのオプション」](#)
- ◆ [139 ページの「LDAP ソースハンドラのオプション」](#)
- ◆ [141 ページの「LDAP ターゲットハンドラのオプション」](#)
- ◆ [142 ページの「DELIM ソースハンドラのオプション」](#)
- ◆ [143 ページの「DELIM ターゲットハンドラのオプション」](#)
- ◆ [144 ページの「SCH ソースハンドラのオプション」](#)
- ◆ [144 ページの「LOAD ソースハンドラのオプション」](#)

## 一般オプション

一般オプションは、Novell インポート / エクスポート変換エンジンの処理全体に影響のあるオプションです。

オプション	説明
-C	スキーマキャッシュを使用して比較および更新を実行する場合に指定します。
-l <i>log_file</i>	出力メッセージ(エラーメッセージなど)を書き込むログファイルの名前を指定します。このオプションを指定しなかった場合、エラーメッセージは ice.log に出力されます。  Linux、Solaris、AIX、および HP-UX の各システムでは、このオプションを指定しなかった場合、エラーメッセージのログは記録されません。
-o	既存のログファイルを上書きする場合に指定します。このフラグを設定しなかった場合、メッセージは既存のログファイルの末尾に追加されます。
-e <i>LDIF_error_log_file</i>	正常に処理されなかったエントリを書き込むファイルの名前を指定します。エントリは LDIF 形式で書き込まれます。このファイルは、内容を調べてエラーを修正したうえで元のディレクトリに再適用できます。
-p <i>URL</i>	インポート / エクスポート変換エンジンが使用する XML 配置ルールが格納されている場所を指定します。配置ルールはエントリの位置を変更するときに使用します。詳細については、 <a href="#">151 ページの「変換ルール」</a> を参照してください。
-c <i>URL</i>	インポート / エクスポート変換エンジンが使用する XML 作成ルールが格納されている場所を指定します。作成ルールは、インポート時にエントリを正しく作成するために必要な情報が欠落している場合にそれを補うために使用します。詳細については、 <a href="#">151 ページの「変換ルール」</a> を参照してください。
-s <i>URL</i>	インポート / エクスポート変換エンジンが使用する XML スキーママッピングルールが格納されている場所を指定します。スキーママッピングルールによって、転送元サーバのスキーマエレメントを、転送先サーバ上の同等ではあるが異なるスキーマエレメントにマッピングできます。  詳細については、 <a href="#">151 ページの「変換ルール」</a> を参照してください。
-b (NetWare のみ)	実行終了時の ICE コンソール画面で、入力を待って停止しない場合に指定します。
-h または -?	コマンドラインのヘルプを表示します。

## スキーマオプション

スキーマオプションでは、スキーマキャッシュを使用してスキーマの比較および更新操作を実行できます。

オプション	説明
-C -a	ターゲットスキーマを更新します(足りないスキーマを追加します)。
-C -c <i>filename</i>	指定したファイルにターゲットスキーマを出力します。
-C -n	スキーマの事前チェックを無効にします。

## ソースハンドラのオプション

ソースハンドラオプション (-S) で、インポートするデータのソースを決定します。コマンドラインには、次のうちの 1 つだけを指定できます。

オプション	説明
-SLDIF	LDIF ファイルをソースとして指定します。  サポートされている LDIF オプションのリストについては、 <a href="#">138 ページの「LDIF ソースハンドラのオプション」</a> を参照してください。
-SLDAP	LDAP サーバをソースとして指定します。  サポートされている LDAP オプションのリストについては、 <a href="#">139 ページの「LDAP ソースハンドラのオプション」</a> を参照してください。
-SDELIM	コンマ区切りのデータファイルをソースとして指定します。  サポートされている DELIM オプションのリストについては、 <a href="#">142 ページの「DELIM ソースハンドラのオプション」</a> を参照してください。
-SSCH	スキーマファイルをソースとして指定します。  サポートされている SCH オプションのリストについては、 <a href="#">144 ページの「SCH ソースハンドラのオプション」</a> を参照してください。
-SLOAD	DirLoad テンプレートをソースとして指定します。  サポートされている LOAD オプションのリストについては、 <a href="#">144 ページの「LOAD ソースハンドラのオプション」</a> を参照してください。

## ターゲットハンドラのオプション

ターゲットハンドラオプション (-D) で、エクスポートするデータの書き込み先を決定します。コマンドラインには、次のうちの 1 つだけを指定できます。

オプション	説明
-DLDIF	LDIF ファイルを書き込み先として指定します。  サポートされているオプションのリストについては、 <a href="#">138 ページの「LDIF ターゲットハンドラのオプション」</a> を参照してください。
-DLdap	LDAP サーバを書き込み先として指定します。  サポートされているオプションのリストについては、 <a href="#">141 ページの「LDAP ターゲットハンドラのオプション」</a> を参照してください。
-DDELIM	コンマ区切りのデータファイルを書き込み先として指定します。  サポートされているオプションのリストについては、 <a href="#">143 ページの「DELIM ターゲットハンドラのオプション」</a> を参照してください。

## LDIF ソースハンドラのオプション

LDIF ソースハンドラは、LDIF ファイルからデータを読み込んで、それを Novell インポート / エクスポート 変換エンジンに送ります。

オプション	説明
-f <i>LDIF_file</i>	LDIF ソースハンドラで読み込んだ LDIF レコードを格納するファイルの名前を指定します。これらの LDIF レコードはインポート / エクスポート変換エンジンに送信されます。  Linux、Solaris、AIX、および HP-UX の各システムでは、このオプションを指定しなかった場合、入力データは stdin から読み込まれます。
-a	このオプションを設定した場合、指定した LDIF ファイル内のレコードが内容レコード (変更タイプが設定されていないレコード) であれば、これらのレコードの変更タイプは「追加」であるとみなされます。
-c	エラーが発生しても LDIF ソースハンドラの処理を続行したい場合に指定します。ここでいうエラーとは、LDIF 解析エラーや、ターゲットハンドラからのエラー返信などです。このオプションが設定されている場合にエラーが発生すると、LDIF ソースハンドラは、エラーを報告したうえで、LDIF ファイルの次のレコードを検出し、処理を続行します。
-n	実際の更新は行わず、更新を行った場合の結果を印刷して確認する場合に指定します。このオプションを設定すると、LDIF ソースハンドラは、LDIF ファイルの解析は行いますが、Novell インポート / エクスポート変換エンジンやターゲットハンドラへのレコードの送信は行いません。
-m	このオプションを設定した場合、指定した LDIF ファイル内のレコードが内容レコード (変更タイプが設定されていないレコード) であれば、これらのレコードの変更タイプは「変更」であるとみなされます。
-x	このオプションを設定した場合、指定した LDIF ファイル内のレコードが内容レコード (変更タイプが設定されていないレコード) であれば、これらのレコードの変更タイプは「削除」であるとみなされます。
-R <i>value</i>	処理するレコードの範囲を指定します。
-v	ハンドラの冗長モードを有効にします。

## LDIF ターゲットハンドラのオプション

LDIF ターゲットハンドラは、Novell インポート / エクスポート変換エンジンからデータを受け取り、そのデータを LDIF ファイルに書き込みます。

オプション	説明
-f <i>LDIF_file</i>	LDIF レコードの書き込み先になるファイルの名前を指定します。  Linux、Solaris、AIX、および HP-UX の各システムでは、このオプションを指定しなかった場合、出力データは stdout に送られます。
-B	バイナリ値も印刷します。
-b	LDIF データの Base64 エンコードを行いません。

## LDAP ソースハンドラのオプション

LDAP ソースハンドラは、検索リクエストを該当する LDAP サーバに送信することによってそのサーバからデータを読み込みます。LDAP ソースハンドラは、検索操作の結果として受け取った検索エントリを Novell インポート / エクスポート 変換エンジンに送ります。

オプション	説明
-s <i>server_name</i>	ハンドラが検索リクエストを送るときの送信先 LDAP サーバの DNS 名または IP アドレスを指定します。デフォルトはローカルホストです。
-p <i>port</i>	<i>server_name</i> で指定した LDAP サーバのポート番号を整数で指定します。デフォルトは 389 です。セキュリティ保護された処理の場合、デフォルトポートは 636 です。
-d <i>DN</i>	サーバによって指定されたバインド操作に使用するエントリの識別名を指定します。
-w <i>password</i>	<i>DN</i> で指定したエントリのパスワード属性を指定します。
-W	<i>DN</i> で指定したエントリのパスワードの入力を求めるプロンプトが表示されます。 このオプションは、Linux、Solaris、AIX、および HP-UX にのみ適用されます。
-F <i>filter</i>	RFC 1558 準拠の検索フィルタを指定します。このオプションを指定しなかった場合、デフォルトの検索フィルタである「objectclass=*」が使用されます。
-n	実際の検索は行わず、検索の条件などを表示して確認する場合に指定します。
-a <i>attribute_list</i>	検索対象の属性をコンマ区切りのリスト形式で指定します。属性名を指定するか、次の 3 つのうちいずれかを指定します。 <ul style="list-style-type: none"><li>• 属性を検索しない場合は、「1.1」</li><li>• すべてのユーザ属性を検索する場合は、「*」</li><li>• オペレーショナルでない属性をすべて検索する場合は、空のリスト。</li></ul> このオプションを指定しなかった場合、属性リストは空のリスト ( デフォルト ) になります。
-o <i>attribute_list</i>	LDAP サーバから受け取った検索結果をインポート / エクスポート変換エンジンに送信する前に、その検索結果から削除する属性をコンマ区切りのリスト形式で指定します。このオプションは、-a オプションでワイルドカードを指定してクラスの属性を初めにすべて検索してから、その検索結果をインポート / エクスポート変換エンジンに渡す前に一部の属性を削除したい場合に便利です。  たとえば、「-a* -o telephoneNumber」と指定すると、ユーザレベルの属性がすべて検索された後で、その検索結果から telephoneNumber 属性が削除されます。
-R	参照結果を自動的に適用しない場合に指定します。デフォルトでは、-d および -w オプションで指定された名前とパスワードによる参照結果が自動的に適用されます。

オプション	説明
-e <i>value</i>	LDAP クライアント SDK で有効にするデバッグフラグを指定します。  詳細については、511 ページの「LDAP SDK デバッグフラグを使用する」を参照してください。
-b <i>base_DN</i>	検索リクエストのベース識別名を指定します。このオプションを指定しなかった場合、デフォルトのベースDNである""(空の文字列)が使用されます。
-c <i>search_scope</i>	検索リクエストの範囲を指定します。有効な値は次のとおりです。 <ul style="list-style-type: none"> <li>One: この値を指定すると、ベースオブジェクトの直接の子オブジェクトだけが検索の対象になります。</li> <li>Base: この値を指定すると、ベースオブジェクトのエントリだけが検索の対象になります。</li> <li>Sub: この値を指定すると、ベースオブジェクトをルートとする LDAP サブツリー (ベースオブジェクトも含まれる) が検索の対象になります。</li> </ul> このオプションを指定しなかった場合は、検索範囲のデフォルトの「Sub」が使用されます。
-r <i>deref_aliases</i>	検索時に別名を逆参照する方法を指定します。指定できる値は次のとおりです。 <ul style="list-style-type: none"> <li>Never: この値を指定すると、サーバは別名の逆参照を行いません。</li> <li>Always: この値を指定すると、検索のベースオブジェクトを検索するときと検索フィルタに一致するエントリを評価するときの両方で、別名の逆参照が行われます。</li> <li>Search: この値を指定すると、ベースオブジェクトを検出した後に検索範囲内のエントリにフィルタを適用するときには別名の逆参照が行われますが、ベースオブジェクト自体の検索時には別名の逆参照は行われません。</li> <li>Find: この値を指定すると、検索のベースオブジェクトを検索するときには別名の逆参照が行われますが、検索フィルタに一致するエントリを評価するときには別名の逆参照は行われません。</li> </ul> このオプションを指定しなかった場合は、別名逆参照の方式のデフォルトである「Never」が使用されます。
-l <i>time_limit</i>	検索の制限時間を秒単位で指定します。
-z <i>size_limit</i>	検索結果として取得できるエントリの最大数を指定します。
-V <i>version</i>	接続に使用する LDAP プロトコルのバージョンを指定します。2 または 3 を指定します。このオプションを指定しなかった場合のデフォルト値は 3 です。
-v	ハンドラの冗長モードを有効にします。
-L <i>filename</i>	SSL 認証に使用するサーバキーが格納されている DER 形式のファイルを指定します。
-A	属性名だけを取得したいときに指定します。このオプションを指定した場合、出力される検索結果に属性値は含まれません。
-t	エラーが発生しても LDAP ハンドラの処理を続行したい場合に指定します。



オプション	説明
-m	LDAP 操作は「変更」になります。
-x	LDAP 操作は「削除」になります。
-k	SSL を使用して接続します。
-M	Manage DSA IT コントロールを有効にします。
-MM	Manage DSA IT コントロールを有効にし、重要度を高く設定します。

## LDAP ターゲットハンドラのオプション

LDAP ターゲットハンドラは、Novell インポート / エクスポート変換エンジンからデータを受け取り、それを LDAP サーバに送信します。データは更新操作の形で送信され、送信先サーバによって実行されます。

LDIF ファイル内のハッシュ化パスワードについては、[505 ページの「LDIF ファイル内でのハッシュ化パスワードの表記」](#)を参照してください。

オプション	説明
-s <i>server_name</i>	ハンドラが検索リクエストを送るときの送信先 LDAP サーバの DNS 名または IP アドレスを指定します。デフォルトはローカルホストです。
-p <i>port</i>	<i>server_name</i> で指定した LDAP サーバのポート番号を整数で指定します。デフォルトは 389 です。セキュリティ保護された処理の場合、デフォルトポートは 636 です。
-d <i>DN</i>	サーバによって指定されたバインド操作に使用するエントリの識別名を指定します。
-w <i>password</i>	<i>DN</i> で指定したエントリのパスワード属性を指定します。
-W	<i>DN</i> で指定したエントリのパスワードの入力を求めるプロンプトが表示されず。 このオプションは、Linux、Solaris、AIX、および HP-UX にのみ適用されます。
-B	サーバへの更新操作の転送に非同期 LBURP(LDAP Bulk Update/Replication Protocol) リクエストを使用しない場合は、このオプションを指定します。標準の同期 LDAP 更新操作リクエストを使用します。  詳細については、 <a href="#">160 ページの「LBURP(LDAP Bulk Update/Replication Protocol)」</a> を参照してください。
-F	前方参照を作成可能にしたい場合は、このオプションを指定します。作成するエントリのペアレントが存在しない場合、エントリが正常に作成できるよう、そのエントリのペアレントに対応するプレースホルダが作成されます。このプレースホルダを「前方参照」といいます。後の処理でペアレントが作成されると、前方参照は通常のエントリに変更されます。

オプション	説明
-l	パスワード値の格納に NMAS™(Novell Modular Authentication Service) の簡易パスワード方式を使用する場合は、このオプションを指定します。簡易パスワード方式の場合、パスワードはディレクトリ内の安全な場所で保持されますが、キーのペアはサーバ間での認証で実際に必要になるまで生成されません。これにより、パスワード情報を持つオブジェクトのロードにかかる時間を短縮できます。
-e <i>value</i>	LDAP クライアント SDK で有効にするデバッグフラグを指定します。  詳細については、 <a href="#">511 ページの「LDAP SDK デバッグフラグを使用する」</a> を参照してください。
-V <i>version</i>	接続に使用する LDAP プロトコルのバージョンを指定します。2 または 3 を指定します。このオプションを指定しなかった場合のデフォルト値は3です。
-L <i>filename</i>	SSL 認証に使用するサーバキーが格納されている DER 形式のファイルを指定します。
-k	SSL を使用して接続します。
-M	Manage DSA IT コントロールを有効にします。
-MM	Manage DSA IT コントロールを有効にし、重要度を高く設定します。

## DELIM ソースハンドラのオプション

DELIM ソースハンドラはコンマ区切りのデータファイルからデータを読み込み、それをターゲットハンドラに送信します。

オプション	説明
-f <i>filename</i>	DELIM ソースハンドラによって読み込まれるコンマ区切りのレコードを含んだファイルの名前を指定します。これらの DELIM レコードはターゲットハンドラに送信されます。
-F <i>value</i>	-f で指定したファイルに対する属性のデータオーダを含んだファイル名を指定します。このオプションが指定されていない場合、-t を使用してこの情報を直接入力する必要があります。  詳細については、 <a href="#">148 ページの「コンマ区切りのインポートを実行する」</a> を参照してください。
-t <i>value</i>	コンマ区切りの属性リストです。このリストによって、-f で指定されたファイルに対する属性のデータオーダを指定します。このオプションまたは -F オプションのいずれかを指定する必要があります。  詳細については、 <a href="#">148 ページの「コンマ区切りのインポートを実行する」</a> を参照してください。
-c	エラーが発生しても DELIM ソースハンドラの処理を続行したい場合に指定します。ここでいうエラーとは、コンマ区切りデータファイルの解析エラーや、ターゲットハンドラからのエラー返信などです。このオプションが設定されている場合にエラーが発生すると、DELIM ソースハンドラは、エラーを報告したうえで、コンマ区切りデータファイル内の次のレコードを検出し、処理を続行します。

オプション	説明
-n value	新しいオブジェクトに LDAP ネーミング属性を指定します。この属性は、-F または -t を使用して指定する属性データに含まれている必要があります。
-l value	RDN の追加先のパスを指定します (o=myCompany など)。DN を渡す場合は、この値は必要ありません。
-o value	オブジェクトクラスのコンマ区切りのリスト ( 入力ファイルに含まれていない場合 )、または補助クラスなどその他のオブジェクトクラスを指定します。デフォルト値は「inetorgperson」です。
-i value	スキップする列のコンマ区切りのリストです。この値には、スキップする列の数を整数で指定します。たとえば 3 列目と 5 列目をスキップする場合は、「i3,5」と指定します。
-d value	区切り記号を指定します。デフォルトの区切り記号はコンマ (,) です。 次に示すのは、特別な場合の区切り記号です。 [q] = 引用符 ( 二重引用符 " ) 1 つによる区切り ) [t] = タブ たとえば、タブを区切り記号として指定するには -d[t] を渡します。
-q value	セカンダリ区切り記号を指定します。デフォルトのセカンダリ区切り記号は一重引用符 ( ' ) です。 次に示すのは、特別な場合の区切り記号です。 [q] = 引用符 ( 二重引用符 " ) 1 つによる区切り ) [t] = タブ たとえば、タブを区切り記号として指定するには -d[t] を渡します。
-v	冗長モードで実行します。

## DELIM ターゲットハンドラのオプション

DELIM ターゲットハンドラはソースハンドラからデータを受け取り、そのデータをコンマ区切りのデータファイルに書き込みます。

オプション	説明
-f filename	コンマ区切りレコードの書き込み先になるファイルの名前を指定します。
-F value	ソースデータの属性のデータオーダを含んだファイルの名前を指定します。このオプションが指定されていない場合、-t を使用してこの情報を直接入力する必要があります。
-t value	コンマ区切りの属性リストです。このリストによって、ソースデータの属性のデータオーダを指定します。このオプションまたは -F オプションのいずれかを指定する必要があります。
-l value	RDN または DN のいずれかになります。ドライバによってデータ内に配置されるのが、DN 全体または RDN だけのどちらであるかを指定します。デフォルト値は RDN です。

オプション	説明
-d <i>value</i>	区切り記号を指定します。デフォルトの区切り記号はコンマ (,) です。  次に示すのは、特別な場合の区切り記号です。  [q] = 引用符 (二重引用符「"」1つによる区切り) [t] = タブ  たとえば、タブを区切り記号として指定するには -d[t] を渡します。
-q <i>value</i>	セカンダリ区切り記号を指定します。デフォルトのセカンダリ区切り記号は一重引用符 (') です。  次に示すのは、特別な場合の区切り記号です。  [q] = 引用符 (二重引用符「"」1つによる区切り) [t] = タブ  たとえば、タブを区切り記号として指定するには -d[t] を渡します。
-n <i>value</i>	インポート処理中に追加されるネーミング属性を指定します。たとえば cn などです。

### SCH ソースハンドラのオプション

SCH ハンドラは、古い NDS や eDirectory のスキーマファイル ( 拡張子 \*.sch がついたファイル ) からデータを読み込んで、それを Novell インポート / エクスポート変換エンジンに送ります。このハンドラを使用すれば、拡張子 \*.sch がついたファイルを入力として、スキーマ関連の操作を LDAP サーバに実装できます。

SCH ハンドラは、ソースハンドラだけのハンドラです。LDAP サーバに \*.sch ファイルをインポートするときには、スキーマハンドラを使用できます。ただし、\*.sch ファイルはエクスポートできません。

SCH ハンドラでサポートされているオプションを次の表に示します。

オプション	説明
-f <i>filename</i>	*.sch ファイルの完全なパス名を指定します。
-c	( オプション ) エラーが発生しても SCH ハンドラの処理を続行したい場合に指定します。
-v	( オプション ) 冗長モードで実行します。

### LOAD ソースハンドラのオプション

DirLoad ハンドラはテンプレートのコマンドから eDirectory 情報を生成します。このテンプレートファイルは -f 引数で指定されます。また、属性仕様の情報とプログラム制御の情報が保持されています。

オプション	説明
-f <i>filename</i>	すべての属性仕様とプログラムの実行を制御するすべての情報を含んだテンプレートファイルを指定します。

オプション	説明
-c	エラーが通知された場合、次のレコードから続行します。
-v	冗長モードで実行します。
-r	データが追加されずに削除されるように、要求を削除要求に変更します。このオプションにより、DirLoad テンプレートを使用して追加されたレコードを削除できます。
-m	テンプレートファイル内に変更要求を作成します。

**属性仕様** 新しいオブジェクトのコンテキストを決定します。

次の属性仕様ファイルのサンプルを参照してください。

```
givenname:$R(first)
initial:$R(initial)
sn:$R(last)
dn:cn=$A(givenname,%1s)$A(initial,%1s)$A(sn),ou=dev,ou=ds,o=novell
objectclass:inetorgperson
telephonenumber:1-800-$N(1-999,%03d)-$C(%04d)
title:$R(titles)
locality:Our location
```

属性仕様ファイルの形式は LDIF ファイルに似ていますが、属性仕様ファイルでは強力な構成体を使用して、詳細な情報と属性間の関係を指定することができます。

**固有の数値** 指定されたオブジェクトに対する固有の数値を属性値に挿入します。

**構文** :\$C[(<format)]

オプションの <format は、値に適用された出力形式を指定します。形式を指定しない場合、カッコは 2 種類とも必要ありません。

```
$C
$C(%d)
$C(%04d)
```

\$C だけを指定すると、現在の数値を属性値に挿入します。「%d」は、何も指定されなかった場合にプログラムが使用するデフォルト形式であるため、\$C は \$C(%d) と同じです。数値は各オブジェクトの後で増加するため、属性仕様で \$C を複数回使用しても、単一オブジェクト内では数値は変わりません。開始値は、!COUNTER=value の構文を使用して、設定ファイル内で指定できます。

**任意の数値** 次の構文を使用して、属性値に任意の数値を挿入します。

**構文** :\$N(<low-<high[,<format]])

<low と <high では、下限値と上限値を指定します。任意の数字が生成される際に各々の値を使用します。オプションの <format は、値に適用された出力形式を指定します。

```
$N(1-999)
$N(1-999,%d)
$N(1-999,%03d)
```

**リストの任意の文字列** 次の構文を使用して、指定したリストから任意に選択された文字列を属性値に挿入します。

**構文** :\$R(<filename[,<format]])

<filename には、値を格納しているファイルを指定します。ファイルへのパスは絶対パスまたは相対パスのどちらでも指定できます。リストを格納しているファイルには、このパッケージに含まれているものがあります。値は改行文字で区切る必要があります。

オプションの <format は、値に適用された出力形式を指定します。

```
$A(givenname)
$A(givenname,%s)
$A(givenname,%.1s)
```

前方参照は使用できませんので注意してください。属性値を使用する場合、その属性は全て、属性仕様ファイル内で現在の属性より前にある必要があります。次の例では、dn の一部としての cn が givenname、initial、sn から構成されています。したがって、givenname、initial、sn の属性は属性仕様ファイル内では dn の前にある必要があります。

```
givenname: $R(first)
initial: $R(initial)
sn: $R(last)
dn:o=novell,ou=dev,ou=ds,cn=$A(givenname,%.1s)$A(initial,%.1s)$A(sn)
```

dn は LDIF ファイル内では次のような特別な処理を受けます。dn がどこに設定されているかに関係なく、dn は最初に (LDIF 構文どおりに) LDIF ファイルに書き込まれます。その他のすべての属性は、表示された順に書き込まれます。

**制御設定** オブジェクト作成の際の制御を追加します。すべての制御には、属性設定と区別するために、行頭の文字として感嘆符 (!) が付いています。制御はファイル内の任意の場所に置くことができます。

```
!COUNTER=300
!OBJECTCOUNT=2
!CYCLE=title
!UNICYCLE=first,last
!CYCLE=ou,BLOCK=10
```

- ◆ カウンタ

固有のカウンタ値の開始値を提供します。カウンタ値は、\$C 構文内の任意の属性に挿入されます。

- ◆ オブジェクト数

OBJECTCOUNT は、テンプレートから作成されるオブジェクトの数を決定します。

- ◆ サイクル

CYCLE は、ファイル (\$R 構文) から任意の値を抜き出す方法を変更するときに使用できます。この設定には異なる 3 つの値があります。

```
!CYCLE=title
```

「title」というリスト名が使用される場合は常に、値が任意で選択されるのではなく、リストの順に次の値が抜き出されます。順番に値がすべて使用された場合には、再度リストの最初から開始します。

```
!CYCLE=ou,BLOCK=10
```

リスト「ou」のそれぞれの値が 10 回ずつ使用され、その後次の値に移動します。

CYCLE 制御設定のうちで最も興味深い変数は、UNICYCLE です。UNICYCLE は、一連のソースを左から右の順序で繰り返すように指定します。このため、必要な場合に一意の値が必ず作成されます。UNICYCLE 制御を使用する場合、OBJECTCOUNT 制御は、オブジェクト数を、リストから作成できる一意のオブジェクトの最大数に制限するためだけに使用します。つまり、UNICYCLE に含まれるリストが 15000 オ

プロジェクトを作成できる場合、その数を減らすために OBJECTCOUNT を使用することはできますが、増やすために使用することはできません。

たとえば、givenname ファイルが 2 つの値 (「Doug」および「Karl」) を格納しており、かつ sn ファイルが 3 つの値 (「Hoffman」、「Schultz」、および「Grieger」) を格納している場合、制御設定「!UNICYCLE=givenname,sn」および cn の属性定義は次のようになります。\$R(givenname) \$R(sn)。ここでは次の cn が作成されます。

```
cn: Doug Hoffman
cn: Karl Hoffman
cn: Doug Schultz
cn: Karl Schultz
cn: Doug Grieger
cn: Karl Grieger
```

## 例

ここでは、Novell インポート / エクスポート変換ユーティリティのコマンドラインユーティリティで次の操作を行う場合のコマンドの例を紹介します。

- ◆ 147 ページの「LDIF インポートを実行する」
- ◆ 147 ページの「LDIF エクスポートを実行する」
- ◆ 148 ページの「コンマ区切りのインポートを実行する」
- ◆ 148 ページの「コンマ区切りのエクスポートを実行する」
- ◆ 148 ページの「LDAP サーバ間でデータを移行する」
- ◆ 148 ページの「スキーマのインポートを実行する」
- ◆ 149 ページの「LOAD ファイルのインポートを実行する」

### LDIF インポートを実行する

LDIF のインポートを実行するには、LDIF ソースハンドラと LDAP ターゲットハンドラを組み合わせて次の例のように指定します。

```
ice -S LDIF -f entries.ldif -D LDAP -s server1.acme.com -p 389 -d
cn=admin,c=us -w secret
```

コマンドラインでこのように指定すると、LDIF データが entries.ldif から読み込まれ、ポート 389 にある LDAP サーバ server1.acme.com に送られます。送信時の識別子は「cn=admin,c=us」、パスワードは「secret」になります。

### LDIF エクスポートを実行する

LDIF のエクスポートを実行するには、LDAP ソースハンドラと LDIF ターゲットハンドラを組み合わせます。次に例を示します。

```
ice -S LDAP -s server1.acme.com -p 389 -d cn=admin,c=us -w password -F
objectClass=* -c sub -D LDIF -f server1.ldif
```

コマンドラインでこのように指定すると、識別子「cn=admin,c=us」およびパスワード「password」を使用してサブツリー検索が実行され、ポート 389 にある LDAP サーバ server1.acme.com 内のオブジェクトがすべて検出されます。結果のデータは LDIF 形式で server1.ldif に出力されます。

## コンマ区切りのインポートを実行する

コンマ区切りのインポートを実行するには、コマンドを次の例のように指定します。

```
ice -S DELIM -f/tmp/in.csv -F /tmp/order.csv -ncn -lo=acme -D LDAP -s
server1.acme.com -p389 -d cn=admin,c=us -w secret
```

コマンドをこのように指定すると、`/tmp/in.csv` ファイルからコンマ区切りの値が読み込まれ、`/tmp/order.csv` ファイルから属性の順序が読み込まれます。`in.csv` 内の各属性エントリに対して、`order.csv` で属性タイプが指定されます。たとえば、`in.csv` に次の値があるとします。

```
pat,pat,engineer,john
```

この場合、`order.csv` に含まれる値は次のようになります。

```
dn,cn,title,sn
```

`order.csv` の情報は、`-t` オプションを使用して直接指定することもできます。

次にデータは、識別子「`cn=admin,c=us`」、およびパスワード「`secret`」を使用して、ポート 389 で LDAP サーバ `server1.acme.com` に送られます。

この例では、`-n` オプションを使用して、`cn` がオブジェクトの新しい DN になるように指定し、`-l` オプションを使用して、このオブジェクトが組織コンテナ `acme` に追加されるようにします。

## コンマ区切りのエクスポートを実行する

コンマ区切りのエクスポートを実行するには、コマンドを次の例のように指定します。

```
ice -S LDAP -s server1.acme.com -p 389 -d cn=admin,c=us -w password -l
objectClass=* -c sub -D DELIM -f /tmp/server1.csv -F order.csv
```

コマンドラインでこのように指定すると、識別子「`cn=admin,c=us`」およびパスワード「`password`」を使用してサブツリー検索が実行され、ポート 389 にある LDAP サーバ `server1.acme.com` 内のオブジェクトがすべて検出されます。結果のデータはコンマ区切りの形式で `/tmp/server1.csv` ファイルに出力されます。

## LDAP サーバ間でデータを移行する

LDAP サーバ間でデータを移行するには、LDAP ソースハンドラと LDAP ターゲットハンドラを組み合わせます。次に例を示します。

```
ice -S LDAP -s server1.acme.com -p 389 -d cn=admin,c=us -w password -F
objectClass=* -c sub -D LDAP -s server2.acme.com -p 389 -d cn=admin,c=us -w
secret
```

コマンドラインでこのように指定すると、識別子「`cn=admin,c=us`」およびパスワード「`password`」を使用してサブツリー検索が実行され、ポート 389 にある LDAP サーバ `server1.acme.com` 内のオブジェクトがすべて検出されます。結果のデータは、識別子「`cn=admin,c=us`」およびパスワード「`secret`」で、ポート 389 にある LDAP サーバ `server2.acme.com` に送られます。

## スキーマのインポートを実行する

スキーマファイルのインポートを実行するには、コマンドを次の例のように指定します。

```
ice -S SCH -f $HOME/myfile.sch -D LDAP -s myserver -d cn=admin,o=novell -w
passwd
```



コマンドラインでこのように指定すると、スキーマデータが `myfile.sch` から読み込まれ、LDAP サーバ「`myserver`」に送られます。送信時の識別子は「`cn=admin,o=novell`」、パスワードは「`passwd`」になります。

## LOAD ファイルのインポートを実行する

LOAD ファイルのインポートを実行するには、コマンドを次の例のように指定します。

```
ice -S LOAD -f attrs -D LDIF -f new.ldf
```

次に、属性ファイル「`attrs`」の内容の例を示します。

```
#=====
# DirLoad 1.00
#=====

!COUNTER=300
!OBJECTCOUNT=2
#-----

# ATTRIBUTE TEMPLATE
# -----

objectclass: inetorgperson
givenname: $R(first)
initials: $R(initial)
sn: $R(last)
dn: cn=$A(givenname,%1s)$A(initial,%1s)$A(sn),ou=$R(ou),ou=dev,o=novell,
telephonenumber: 1-800-$N(1-999,%03d)-$C(%04d)
title: $R(titles)
```

コマンドプロンプトから前のコマンドを実行すると、次のLDIFファイルが作成されます。

```
version: 1
dn: cn=JohnBBill,ou=ds,ou=dev,o=novell
changetype: add
objectclass: inetorgperson
givenname: John
initials: B
sn: Bill
telephonenumber: 1-800-290-0300
title: Amigo

dn: cn=BobJAmy,ou=ds,ou=dev,o=novell
```

```
changetype: add
objectclass: inetorgperson
givenname: Bob
initials: J
sn: Amy
telephonenumber: 1-800-486-0301
title: Pomo
```

コマンドプロンプトから次のコマンドを実行すると、データが LDAP ハンドラを經由して LDAP サーバに送られます。

```
ice -S LOAD -f attrs -D LDAP -s www.novell.com -d cn=admin,o=novell -w admin
```

次のコマンドラインを使用する際に前のテンプレートファイルを使用すると、前のコマンドで追加したすべてのレコードが削除されます。

```
ice -S LOAD -f attrs -r -D LDAP -s www.novell.com -d cn=admin,o=novell -w admin
```

-m を使用して変更する場合は、次の例のようにレコードを変更します。

```
# =====
# DirLoad 1.00
# =====
!COUNTER=300
!OBJECTCOUNT=2
#-----
# ATTRIBUTE TEMPLATE
# -----
dn: cn=$R(first),%.1s) ($R(initial),%.1s)$R(last),ou=$R(ou),ou=dev,o=novell
delete: givenname
add: givenname
givenname: test1
replace: givenname
givenname: test2
givenname: test3
```

「attrs」ファイルが上のデータを格納しているときに次のコマンドラインを使用した場合の例を示します。

```
ice -S LOAD -f attrs -m -D LDIF -f new.ldf
```

LDIF データは次のような結果になります。

```

version: 1
dn: cn=BillTSmith,ou=ds,ou=dev,o=novell
changetype: modify
delete: givenname
-
add: givenname
givenname: test1
-
replace: givenname
givenname: test2
givenname: test3
-
dn: cn=JohnAWilliams,ou=ldap,ou=dev,o=novell
changetype: modify
delete: givenname
-
add: givenname
givenname: test1
-
replace: givenname
givenname: test2
givenname: test3
-

```

## 変換ルール

Novell インポート/エクスポート変換エンジンは、ソースハンドラから受け取ったレコードをターゲットハンドラに送る前に、レコードに対して変換処理を行います。この変換処理の内容は一連のルールを使用して指定できます。これらのルールは XML で記述されます(XMLファイルとして作成される場合と、XML用ディレクトリ内に格納されたXMLデータとして作成される場合があります)。このルールにより、LDAP ディレクトリ間でのエントリのインポート時に、次の問題が解決されます。


- ◆ 不足している情報
- ◆ 階層の違い
- ◆ スキーマの違い

次の 3 種類の変換ルールがあります。

ルール	説明
配置	<p>エントリの位置を変更します。</p> <p>たとえば、あるユーザグループをいったん「l=San Francisco, c=US」というコンテナにインポートし、インポートが終わった後で「l=Los Angeles, c=US」というコンテナに移したい場合などに、配置ルールを利用できます。</p> <p>これらのルールの形式については、<a href="#">157 ページの「配置ルール」</a>を参照してください。</p>
作成	<p>インポート時にエントリを正しく作成するために必要な情報が欠落している場合にそれを補います。</p> <p>たとえば、LDIF データのエクスポート元サーバのスキーマではユーザエントリに必要とされる属性が cn(commonName) 属性だけであるのに対し、LDIF データのインポート先サーバのスキーマでは cn 属性の他に sn(surname) 属性も必要とされる場合が考えられます。このような場合は、作成ルールを使用することにより、インポート / エクスポート変換エンジンが各エントリを処理するときに、そのエントリにデフォルトの sn 値 (" " など) が設定されるようにできます。これにより、各エントリはインポート先サーバに送信されるときには必要な属性である sn 属性を持つことになり、エントリの正常な追加が保証されます。</p> <p>これらのルールの形式については、<a href="#">155 ページの「作成ルール」</a>を参照してください。</p>
スキーママッピング	<p>サーバ間でデータを転送する場合 ( 直接転送するか LDIF を使用するかに関係なく ) で、転送元サーバのスキーマと転送先サーバのスキーマが異なる場合、次のようにスキーママッピングを使用できます。</p> <ul style="list-style-type: none"> <li>• エクスポート元サーバからインポートするエントリのオブジェクトクラスと属性タイプをすべて受け付けることができるように、インポート先サーバ上のスキーマを拡張します。</li> <li>• 転送元サーバのスキーマエレメントを、転送先サーバ上の同等ではあるが異なるスキーマエレメントにマッピングします。</li> </ul> <p>これらのルールの形式については、<a href="#">153 ページの「スキーママッピングルール」</a>を参照してください。</p>

これらの変換ルールは、Novell eDirectory インポート / エクスポート変換ウィザードとコマンドラインインタフェースのどちらでも利用できます。XML ルールの詳細については、[153 ページの「XML ルールを使用する」](#)を参照してください。

### Novell eDirectory インポート / エクスポート変換ウィザードを使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート変換ウィザード] の順にクリックします。
- 3 実行するタスクを選択します。
- 4 [詳細設定] の下の次のオプションから選択します。

オプション	説明
スキーマルール	インポート/エクスポート変換エンジンが使用する XML スキーママッピングルールが格納されている場所を指定します。
配置ルール	インポート/エクスポート変換エンジンが使用する XML 配置ルールが格納されている場所を指定します。
作成ルール	インポート/エクスポート変換エンジンが使用する XML 作成ルールが格納されている場所を指定します。

5 [次へ] をクリックします。

6 表示される指示に従って、選択したタスクを完了します。

## コマンドラインインタフェースを使用する

コマンドラインバージョンで変換ルールを使用するには、Novell インポート/エクスポート変換ユーティリティの実行ファイルを起動するときに、使用したいルールに対応する一般オプション (-p、-c、または -s) を指定します。詳細については、[136 ページの「一般オプション」](#)を参照してください。

オプション	説明
-p URL	インポート/エクスポート変換エンジンが使用する XML 配置ルールが格納されている場所です。
-c URL	インポート/エクスポート変換エンジンが使用する XML 作成ルールが格納されている場所です。
-s URL	インポート/エクスポート変換エンジンが使用する XML スキーママッピングルールが格納されている場所です。

すべての 3 つのオプションで、URL を次のいずれかに指定します。

- ◆ 次の形式の URL:

```
file://[path/]filename
```

ファイルは、ローカルファイルシステムに存在する必要があります。

- ◆ ベースレベルの検索を指定する RFC 2255 準拠の LDAP URL と、1 つの値の属性タイプに対して 1 つの属性記述が定義されている属性リスト。

## XML ルールを使用する

Novell インポート/エクスポート変換ルールで使用される XML 形式は、DirXML<sup>®</sup> の場合と同じです。DirXML の詳細については、『[DirXML Administration Guide \(DirXML 管理ガイド\)](http://www.novell.com/documentation/japanese/dirxml11a/index.html) (<http://www.novell.com/documentation/japanese/dirxml11a/index.html>)』を参照してください。

### スキーママッピングルール

スキーママッピングルールの最上位エレメントは、<attr-name-map> です。インポートスキーマとエクスポートスキーマの相互関係は、マッピングルールによって決まります。マッピングルールは、指定されたインポートクラスの定義や属性を、対応するエクスポートスキーマの定義に関連付けます。

マッピングルールは、属性名またはクラス名に対応させて設定します。

- ◆ 属性マッピングの場合、マッピングルールでは、それが属性マッピングであること、ネームスペース (**nds-name** はソース名のタグ)、**eDirectory** ネームスペース内での名前、および他のネームスペース (**app-name** はターゲット名のタグ) とそのネームスペース内での名前を指定する必要があります。マッピングルールでは、マッピングが特定のクラスに適用されることを指定することも、その属性を持つすべてのクラスに適用されることを指定することもできます。
- ◆ クラスマッピングの場合、マッピングルールでは、それがクラスマッピングルールであること、ネームスペース (**eDirectory** またはアプリケーション) とそのネームスペース内での名前、およびその他のネームスペースとそのネームスペース内での名前を指定する必要があります。

スキーママッピングルールの正式な DTD 定義を次に示します。

```
<!ELEMENT attr-name-map (attr-name | class-name)*>

<!ELEMENT attr-name (nds-name, app-name)>
<!ATTLIST attr-name
      class-name      CDATA      #IMPLIED>

<!ELEMENT class-name (nds-name, app-name)>

<!ELEMENT nds-name (#PCDATA)>

<!ELEMENT app-name (#PCDATA)>
```

複数のマッピングエレメントをファイルに定義できます。各エレメントは、ファイルに定義されている順番で処理されます。1 つのクラスまたは属性を複数回マッピングした場合は、最初のマッピングが優先されます。

スキーママッピングルールの作成例を次に示します。

**スキーマルール 1:** 次のルールでは、**inetOrgPerson** クラスについて、ソースの **surname** 属性をターゲットの **sn** 属性にマッピングします。

```
<attr-name-map>
  <attr-name class-name="inetOrgPerson">
    <nds-name>surname</nds-name>
    <app-name>sn</app-name>
  </attr-name>
</attr-name-map>
```

**スキーマルール 2:** 次のルールでは、ソースの **inetOrgPerson** クラスの定義をターゲットの **User** クラスの定義にマッピングします。

```
<attr-name-map>
  <class-name>
    <nds-name>inetOrgPerson</nds-name>
    <app-name>User</app-name>
  </class-name>
</attr-name-map>
```

**スキーマルール 3:** 次の例では、2 種類のルールを定義します。1 つ目のルールでは、**Surname** 属性を使用するすべてのクラスについて、ソースの **Surname** 属性をターゲットの **sn** 属性にマッピングします。2 つ目のルールでは、ソースの **inetOrgPerson** クラスの定義をターゲットの **User** クラスの定義にマッピングします。

```

<attr-name-map>
  <attr-name>
    <nds-name>surname</nds-name>
    <app-name>sn</app-name>
  </attr-name>
  <class-name>
    <nds-name>inetOrgPerson</nds-name>
    <app-name>User</app-name>
  </class-name>
</attr-name-map>

```

**コマンド例：**スキーマルールが `sr1.xml` ファイルに保存されている場合、次のコマンドを指定することにより、`lentry.ldf` ファイルの処理中にそのルールを使用すること、および結果をターゲットファイル `outt1.ldf` に送ることがインポート / エクスポート変換ユーティリティに指示されます。

```
ice -o -sfile://sr1.xml -SLDIF -flentry.ldf -c -DLDIF
-foutt1.ldf
```

## 作成ルール

作成ルールによって、宛先ディレクトリ内に新規エントリを作成する場合の条件が指定されます。次のエレメントがサポートされます。

- ◆ **必須属性：**すべての必須属性について、追加レコードに値が必要であること、値がない場合には追加が失敗することを指定します。作成ルールでは、必須属性のデフォルト値を指定できません。レコードに属性値がない場合、そのエントリにはデフォルト値が使用されます。レコードに属性値がある場合は、そのレコード値が使用されます。
- ◆ **一致属性：**追加レコードに特定の属性が必要であり、特定の値に一致すること、そうでない場合には追加が失敗することを指定します。
- ◆ **テンプレート：**`eDirectory` 内のテンプレートオブジェクトの識別名を指定します。現時点では、Novell インポート / エクスポート変換ユーティリティの作成ルールにテンプレートを指定することはできません。

作成ルールの正式な DTD 定義を次に示します。

```

<!ELEMENT create-rules (create-rule)*>
<!ELEMENT create-rule (match-attr*,
                        required-attr*,
                        template?)>
<!ATTLIST create-rule
  class-name CDATA #IMPLIED
  description CDATA #IMPLIED>
<!ELEMENT match-attr (value)+ >
<!ATTLIST match-attr
  attr-name CDATA #REQUIRED>
<!ELEMENT required-attr (value)*>
<!ATTLIST required-attr
  attr-name CDATA #REQUIRED>
<!ELEMENT template EMPTY>
<!ATTLIST template
  template-dn CDATA #REQUIRED>

```

複数の作成ルールをファイルに定義できます。各ルールは、ファイルに定義されている順番で処理されます。ルールに適合しないレコードがあると、そのレコードはスキップされますが、レコードのスキップによるエラーは生成されません。

作成ルールの形式例を次に示します。

**作成ルール 1:** 次に紹介するルールでは、`inetOrgPerson` クラスの追加レコードに次の3つの条件が適用されます。追加レコードには、`givenName` 属性および `Surname` 属性が必要です。追加レコードには `L` 属性が必要ですが、この属性値がない場合には、作成ルールによってデフォルト値「`Provo`」に設定されます。

```
<create-rules>
  <create-rule class-name="inetOrgPerson">
    <required-attr attr-name="givenName"/>
    <required-attr attr-name="surname"/>
    <required-attr attr-name="L">
      <value>Provo</value>
    </required-attr>
  </create-rule>
</create-rules>
```

**作成ルール 2:** 次に紹介する作成ルールでは、ベースクラスの種類に関係なく、すべての追加レコードに次の3つの条件が適用されます。

- ◆ 追加レコードには、`givenName` 属性が必要です。この属性値がない場合、追加は失敗します。
- ◆ 追加レコードには、`Surname` 属性が必要です。この属性値がない場合、追加は失敗します。
- ◆ 追加レコードには、`L` 属性が必要です。この属性がない場合、`L` 属性はデフォルト値「`Provo`」に設定されます。

```
<create-rules>
  <create-rule>
    <required-attr attr-name="givenName"/>
    <required-attr attr-name="Surname"/>
    <required-attr attr-name="L">
      <value>Provo</value>
    </required-attr>
  </create-rule>
</create-rules>
```

**作成ルール 3:** 次に紹介する作成ルールでは、ベースクラスの種類に関係なく、すべての追加レコードに次の2つの条件が適用されます。

- ◆ 作成ルールは、レコードに `uid` 属性として `ratuid` が指定されているかチェックします。この属性値がない場合、追加は失敗します。
- ◆ 作成ルールは、レコードに `L` 属性が指定されているかチェックします。この属性がない場合、`L` 属性はデフォルト値「`Provo`」に設定されます。

```
<create-rules>
  <create-rule>
    <match-attr attr-name="uid">
      <value>cn=ratuid</value>
    </match-attr>
    <required-attr attr-name="L">
      <value>Provo</value>
    </required-attr>
  </create-rule>
</create-rules>
```



**コマンド例:** 作成ルールを `cr1.xml` ファイルに保存し、次のコマンドを指定することにより、`lentry.ldf` ファイルの処理中にそのルールを使用すること、および結果をターゲットファイル `outt1.ldf` に送ることがインポート / エクスポート変換ユーティリティに指示されます。

```
ice -o -cfile://cr1.xml -SLDIF -flentry.ldf -c -DLIDIF
-foutt1.ldf
```

## 配置ルール

配置ルールによって、ターゲットディレクトリ内でエントリが作成される位置が決まります。配置ルールでは、次の3つの条件を使用して、エントリの配置にそのルールを適用すべきかどうかを決定します。

- ◆ **クラス一致:** 配置ルールに `match class` エlementが定義されている場合、レコードに定義されている `objectClass` は、ルールの `class-name` 属性に一致する必要があります。一致しない場合、そのレコードには配置ルールが使用されません。
- ◆ **属性一致:** 配置ルールに `match attribute` エlementが定義されている場合、レコードでは、`match attribute` エlementに定義されている各属性について属性値が必要です。一致しない場合、そのレコードには配置ルールが使用されません。
- ◆ **パス一致:** 配置ルールに `match path` エlementが定義されている場合、レコードの `dn` 部分は、`match path` エlementに定義されているプリフィックスに一致する必要があります。一致しない場合、そのレコードには配置ルールが使用されません。

ルールの最後のElementによって、エントリの配置場所が決まります。配置ルールでは、必要に応じて次のオプションを指定できます。

- ◆ **解析済み文字データ:** 解析済み文字データを使用して、エントリに使用するコンテナの DN を指定します。
- ◆ **名前をコピー:** 古い DN のネーミング属性を、エントリの新しい DN で使用することを指定します。
- ◆ **属性をコピー:** エントリの新しい DN で使用するネーミング属性を指定します。指定されたネーミング属性は、エントリのベースクラスのネーミング属性として有効でなければなりません。
- ◆ **パスをコピー:** ソース DN をターゲット DN として使用することを指定します。
- ◆ **パスサフィックスをコピー:** ソース DN のパスの一部をターゲット DN として使用することを指定します。`match-path` Elementを指定した場合、古い DN のパスの一部、つまり、`match-path` Elementのプリフィックス属性に一致しない部分だけが、エントリの DN の一部として使用されます。

配置ルールの正式な DTD 定義を次に示します。

```
<!ELEMENT placement-rules (placement-rule*)>
<!ATTLIST placement-rules
    src-dn-format      (%dn-format;)    "slash"
    dest-dn-format     (%dn-format;)    "slash"
    src-dn-delims      CDATA            #IMPLIED
    dest-dn-delims     CDATA            #IMPLIED>

<!ELEMENT placement-rule (match-class*,
    match-path*,
    match-attr*,
    placement)>
<!ATTLIST placement-rule
    description        CDATA            #IMPLIED>
```

```

<!ELEMENT match-class      EMPTY>
<!ATTLIST match-class
      class-name      CDATA      #REQUIRED>

<!ELEMENT match-path      EMPTY>
<!ATTLIST match-path
      prefix          CDATA      #REQUIRED>

<!ELEMENT match-attr      (value)+ >
<!ATTLIST match-attr
      attr-name      CDATA      #REQUIRED>

<!ELEMENT placement
      (#PCDATA |
      copy-name |
      copy-attr |
      copy-path |
      copy-path-suffix)* >

```

複数の配置ルールエレメントをファイルに定義できます。各ルールは、ファイルに定義されている順番で処理されます。ルールに適合しないレコードがあると、そのレコードはスキップされますが、レコードのスキップによるエラーは生成されません。

配置ルールの形式例を次に示します。`scr-dn-format="ldap"` 属性および `dest-dn-format="ldap"` 属性によって、ソース DN およびターゲット DN のネームスペースが LDAP 形式として定義されます。

Novell インポート/エクスポート変換ユーティリティがサポートするソース名およびターゲット名は、LDAP 形式だけです。

**配置例 1:** 次の配置ルールでは、レコードはベースクラス `inetOrgPerson` を持つ必要があります。レコードがこの条件に適合する場合、そのエントリは `test` コンテナの直下に置かれ、ソース DN の最上位コンポーネントがエントリの DN の一部として使用されます。

```

<placement-rules src-dn-format="ldap" dest-dn-format="ldap">
  <placement-rule>
    <match-class class-name="inetOrgPerson"></match-class>
    <placement>cn=<copy-name/>,o=test</placement>
  </placement-rule>
</placement-rules>

```

ベースクラス `inetOrgPerson` および次の DN を持つレコードがあるとします。

```
dn: cn=Kim Jones, ou=English, ou=Humanities, o=UofZ
```

このレコードは、例に示したルールに従って、ターゲットディレクトリ内で次の DN を持ちます。

```
dn: cn=Kim Jones, o=test
```

**配置例 2:** 次の配置ルールでは、レコードは `sn` 属性を持つ必要があります。レコードがこの条件に適合する場合、そのエントリは `test` コンテナの直下に置かれ、ソース DN の最上位コンポーネントがエントリの DN の一部として使用されます。

```

<placement-rules src-dn-format="ldap" dest-dn-format="ldap">
  <placement-rule>
    <match-attr attr-name="sn"></match-attr>
    <placement>cn=<copy-name/>,o=test</placement>
  </placement-rule>
</placement-rules>

```

次の DN および sn 属性を持つレコードがあるとします。

```
dn: cn=Kim Jones, ou=English, ou=Humanities, o=UofZ
sn: Jones
```

このレコードは、例に示したルールに従って、ターゲットディレクトリ内で次の DN を持ちます。

```
dn: cn=Kim Jones, o=test
```

**配置例 3:** 次の配置ルールでは、レコードは sn 属性を持つ必要があります。レコードがこの条件に適合する場合、そのエントリは test コンテナの直下に置かれ、sn 属性がエントリの DN の一部として使用されます。copy-attr エlement に指定された属性は、エントリのベースクラスのネーミング属性でなければなりません。

```
<placement-rules src-dn-format="ldap" dest-dn-format="ldap">
  <placement-rule>
    <match-attr attr-name="sn"></match-attr>
    <placement>cn=<copy-attr attr-name="sn"/>,o=test</placement>
  </placement-rule>
</placement-rules>
```

次の DN および sn 属性を持つレコードがあるとします。

```
dn: cn=Kim Jones, ou=English, ou=Humanities, o=UofZ
sn: Jones
```

このレコードは、例に示したルールに従って、ターゲットディレクトリ内で次の DN を持ちます。

```
dn: cn=Jones, o=test
```

**配置例 4:** 次の配置ルールでは、レコードは sn 属性を持つ必要があります。レコードがこの条件に適合する場合、ソース DN がターゲット DN として使用されます。

```
<placement-rules src-dn-format="ldap" dest-dn-format="ldap">
  <placement-rule>
    <match-attr attr-name="sn"></match-attr>
    <placement><copy-path/></placement>
  </placement-rule>
</placement-rules>
```

**配置例 5:** 次の配置ルールでは、レコードは sn 属性を持つ必要があります。レコードがこの条件に適合する場合、エントリの DN 全体が test コンテナにコピーされます。

```
<placement-rules src-dn-format="ldap" dest-dn-format="ldap">
  <placement-rule>
    <match-attr attr-name="sn"></match-attr>
    <placement><copy-path-suffix/>,o=test</placement>
  </placement-rule>
</placement-rules>
```

次の DN および sn 属性を持つレコードがあるとします。

```
dn: cn=Kim Jones, ou=English, ou=Humanities, o=UofZ
sn: Jones
```

このレコードは、例に示したルールに従って、ターゲットディレクトリ内で次の DN を持ちます。

```
dn: cn=Kim Jones, ou=English, ou=Humanities, o=UofZ, o=test
```

**配置例 6:** 次の配置ルールでは、レコードは `sn` 属性を持つ必要があります。レコードがこの条件に適合する場合、エントリの DN 全体が `neworg` コンテナにコピーされます。

```
<placement-rules>
  <placement-rule>
    <match-path prefix="o=engineering"/>
    <placement><copy-path-suffix/>o=neworg</placement>
  </placement-rule>
</placement-rules>
```

次に例を示します。

```
dn: cn=bob,o=engineering
```

これは次のようになります。

```
dn: cn=bob,o=neworg
```

**コマンド例 :** 配置ルールが `pr1.xml` ファイルに保存されている場合、次のコマンドを指定することにより、`lentry.ldf` ファイルの処理中にそのルールを使用すること、および結果をターゲットファイル `foutt1.ldf` に送ることがインポート / エクスポート変換ユーティリティに指示されます。

```
ice -o -pfile://pr1.xml -SLDIF -flentry.ldf -c -DLDIF
-foutt1.ldf
```

## LBURP(LDAP Bulk Update/Replication Protocol)

Novell インポート / エクスポート変換ユーティリティでは、LDAP サーバへの非同期リクエストの送信に、LBURP を使用します。これにより、リクエストは常にプロトコルで指定された順序で処理されます。複数プロセッサ間の相互関係やオペレーティングシステムのスケジューラの設定によって処理順序が変わることはありません。

LBURP により、Novell インポート / エクスポート変換ユーティリティは、複数の更新操作を 1 つのリクエストとして送信したり、これらすべての更新操作に対する応答を 1 つのレスポンスとして受け取ることができます。これにより、プロトコルのネットワーク処理効率が向上します。

LBURP は次のように機能します。


1. Novell インポート / エクスポート変換ユーティリティが LDAP サーバにバインドします。
2. サーバからクライアントにバインドレスポンスが送られます。
3. クライアントからサーバに開始 LBURP 拡張リクエストが送られます。
4. サーバからクライアントに開始 LBURP 拡張レスポンスが送られます。
5. 必要に応じてクライアントからサーバに LBURP 操作拡張リクエストが送られます。  
これらのリクエストは非同期で送信することもできます。リクエストごとに通し番号が付けられ、同じクライアントから同じ接続を介して送信された個々のリクエストの順序はこの通し番号によって特定されます。各リクエストには、少なくとも 1 つの LDAP 更新操作が設定されます。
6. サーバは、受け取った各 LBURP 操作拡張リクエストを通し番号に従って順番に処理し、リクエストごとに LBURP 操作拡張レスポンスを送信します。
7. サーバへの更新操作の送信がすべて終了すると、クライアントはサーバに終了 LBURP 拡張リクエストを送ります。
8. サーバからクライアントに終了 LBURP 拡張レスポンスが送られます。

LBURP プロトコルにより、Novell インポート / エクスポート 変換機能は、サーバにデータを転送するときに送信元と送信先の間のネットワーク接続の限界まで転送速度を上げることができます。ネットワーク接続が十分に高速であれば、Novell インポート / エクスポート 変換機能からリクエストが送られてくるのを待つ必要がないため、サーバはすべての処理時間を更新操作の処理だけに費やすことができます。

更新操作の処理効率をさらに上げるため、eDirectory の LBURP プロセッサは、データベースへの更新操作をグループに分けて行います。LBURP の採用により、従来の同期処理の場合と比べて、LDIF のインポート処理の効率は大幅に改善されています。

LBURP はデフォルトで有効になっていますが、LDIF のインポート中に無効にすることもできます。

LDIF のインポート中に LBURP の有効 / 無効を切り替えるには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート 変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルからデータをインポート] > [次へ] の順にクリックします。
- 4 [ファイルタイプ] ドロップダウンリストから LDIF を選択し、インポートするデータが格納されている LDIF ファイルの名前を指定します。
- 5 [次へ] をクリックします。
- 6 データをインポートする LDAP サーバとログインのタイプ (匿名ログインまたは認証ログイン) を指定します。
- 7 [詳細設定] の下の [LBURP を使用] を選択します。
- 8 [次へ] をクリックし、表示される指示に従って LDIF インポートウィザードでの残りの作業を完了します。

**重要 :** LBURP は比較的新しいプロトコルであるため、バージョン 8.5 より以前の eDirectory サーバおよび eDirectory 以外のサーバの大部分は、LBURP をサポートしていません。Novell eDirectory インポート / エクスポート 変換ウィザードを使用して LBURP をサポートしていないサーバに LDIF ファイルをインポートする場合は、LDIF のインポートが正しく行われるよう、LBURP オプションを無効にします。

コマンドラインオプションを使用して、LDIF のインポート中に LBURP の有効 / 無効を切り替えることができます。詳細については、[141 ページの「B」](#)を参照してください。

## LDAP ディレクトリ間でスキーマを移行する

LDAP ディレクトリ間でのスキーマの移行に関する詳細については、Novell Developer Portal の『[NetWare Application Notes \(NetWare アプリケーションノート\)](http://www.developer.novell.com/research)』 (<http://www.developer.novell.com/research>) を参照してください。

## LDIF のインポートを高速化する

1 つの LDIF ファイルに数千または数百万のレコードがある場合は、次のことを検討してください。

- ◆ [162 ページの「読み書き可能レプリカを持つサーバに直接インポートする」](#)
- ◆ [162 ページの「LBURP を使用する」](#)
- ◆ [162 ページの「データベースキャッシュを設定する」](#)
- ◆ [162 ページの「簡易パスワードを使用する」](#)
- ◆ [163 ページの「インデックスを使用する場合の注意」](#)

## 読み書き可能レプリカを持つサーバに直接インポートする

実行が可能な場合は、LDIF ファイルで示されているすべてのエントリーを含む、読み書き可能レプリカを持つサーバを LDIF のインポート先に選択します。これによりネットワーク効率を大幅に高めることができます。

更新時には、インポート先サーバから他の eDirectory サーバへのチェーン接続は行わないでください。これにより、パフォーマンスはかなり低下します。ただし、一部の更新対象エントリーが LDAP を実行していないサーバ上だけに存在する場合には、LDIF ファイルをインポートするためにチェーン接続が必要になることもあります。

レプリカとパーティション管理の詳細については、[5 章 117 ページの「パーティションおよびレプリカの管理」](#)を参照してください。

## LBURP を使用する

Novell インポート / エクスポート変換機能では、ネットワークと eDirectory サーバの処理をできるだけ効率化するために、ウィザードとサーバの間でのデータ転送に LBURP を使用します。LDIF のインポート時に LBURP を使用することにより、LDIF のインポートにかかる時間が大幅に短縮されます。

LBURP の詳細については、[160 ページの「LBURP\(LDAP Bulk Update/Replication Protocol\)」](#)を参照してください。

## データベースキャッシュを設定する


eDirectory で使用できるデータベースキャッシュの容量は、LDIF インポートの処理速度に大きく影響します。特に、サーバ上のエントリーの総数が多いほど影響は大きくなります。LDIF のインポートでは、インポート実行中にはできるだけ多くのメモリを eDirectory に割り当てると効率的です。インポートが完了してサーバの負荷が通常レベルに戻ったら、メモリの設定を元にもどすことができます。この方法は、eDirectory サーバで実行する処理がインポートだけの場合に特に効果があります。

eDirectory データベースキャッシュの設定の詳細については、[15 章 443 ページの「Novell eDirectory のメンテナンス」](#)を参照してください。

## 簡易パスワードを使用する

Novell eDirectory では、パブリックキーとプライベートキーのペアを使用して認証を行います。これらのキーの生成は、CPU に大きな負荷のかかる処理です。eDirectory 8.7.3 では、パスワードの格納に、NMASTM(Novell Modular Authentication Service) の簡易パスワード機能を使用できます。簡易パスワードを使用する場合、パスワードはディレクトリ内の安全な場所で保持されますが、キーのペアはサーバ間での認証で実際に必要になるまで生成されません。これにより、パスワード情報を持つオブジェクトをロードする速度を大幅にアップできます。

LDIF のインポート時に簡易パスワードを有効にするには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルからデータをインポート] > [次へ] の順にクリックします。
- 4 [ファイルタイプ] ドロップダウンリストから LDIF を選択し、インポートするデータが格納されている LDIF ファイルの名前を入力します。

- 5 [次へ] をクリックします。
- 6 データをインポートするLDAPサーバとログインのタイプ(匿名ログインまたは認証ログイン)を指定します。
- 7 [詳細設定] の下の [NMASS 通常パスワード / ハッシュ化パスワードを格納する] を選択します。
- 8 [次へ] をクリックし、表示される指示に従って、LDIF インポートウィザードでの残りの作業を完了します。

パスワードの格納に簡易パスワードを使用する場合は、eDirectory ツリーへのログインおよび従来型のファイルサービスや印刷サービスへのアクセスには、NMASS 対応の Novell Client™ を使用する必要があります。またサーバには NMASS がインストールされている必要があります。名前とパスワードのバインドを行う LDAP アプリケーションは、簡易パスワード機能とスムーズに連携します。

NMASS の詳細については、『*Novell Modular Authentication Service Administration Guide(Novell Modular Authentication Service 管理ガイド)* (<http://www.novell.com/documentation/japanese/nmas22/index.html>)』を参照してください。

## インデックスを使用する場合の注意

不要なインデックスがあると、LDIF のインポートにかかる時間が長くなります。これは、定義されているすべてのインデックスで、設定されている属性値を持つエントリごとに追加の処理が実行されるためです。LDIF をインポートする前に、不要なインデックスがないことを確認します。インデックスを作成するときは、あらかじめデータ確認済みのプレディケート統計をロードしてインデックスが本当に必要な箇所を確認すると、不要なインデックスを減らすことができます。

インデックスの調整の詳細については、163 ページの「インデックスマネージャ」を参照してください。

## インデックスマネージャ

インデックスマネージャは、サーバオブジェクトの属性の 1 つで、データベースインデックスの管理に使用します。eDirectory では、データベースインデックスを使用することによって、クエリの処理速度が大幅に向上します。

Novell eDirectory には、基本的なクエリの機能を提供する一連のインデックスが付属しています。これらデフォルトのインデックスの対象となる属性を次に示します。

CN	別名オブジェクト名
dc	Obituary
Given Name	Member
Surname	Reference
uniqueID	Equivalent to Me
GUID	NLS:Common Certificate
cn_SS	Revision
uniqueID_SS	extensionInfo
ldapAttributeList	ldapClassList


またカスタマイズされたインデックスを作成して、ユーザの環境における eDirectory のパフォーマンスをさらに向上させることができます。たとえば、デフォルトでインデックス付けされていない属性を検索する新しい LDAP アプリケーションが組織に導入された場合、その属性に対するインデックスを作成すると便利です。

**注：**インデックスを使用することにより検索の処理速度は上がりますが、インデックスの数が増えるほど更新にかかる時間が長くなります。一般には、パフォーマンスの問題が特定のディレクトリの検索に関係すると思われる場合に、新しいインデックスを作成します。

Novell iManager を使用して、インデックスを作成または削除します。インデックス名、状態、タイプ、ルール、インデックス付き属性など、インデックスのプロパティを表示したり、管理することができます。

プレディケート統計データを使用すると、どのようなインデックスを追加したらユーザの環境で便利であるかを確認できます。プレディケート統計データは ConsoleOne でのみ使用できます。[168 ページの「プレディケートデータ」](#)を参照してください。

## インデックスを作成する

1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。

2 [eDirectory の保守] > [インデックス管理] の順にクリックします。

3 利用可能なサーバのリストからサーバを選択します。

4 [インデックスの変更] ページで [作成] をクリックします。

5 インデックス名を入力します。

インデックス名を入力しなかった場合は、選択した属性が自動的にインデックス名として設定されます。

**重要：**「\$」文字は属性値の区切り記号として使用されます。インデックス名に「\$」文字を使用する場合、前に円記号 (¥) を付けて、LDAP でインデックスを作成するときに「\$」文字をエスケープします。

6 属性を選択します。

7 インデックスのルールを選択します。

- ◆ **値：**属性の値全体または値の最初の部分を照合します。たとえば、値一致は、「Jensen」に一致する「LastName」のあるエントリの検索や、「Jen」で始まる「LastName」があるエントリの検索に使用できます。
- ◆ **存在：**特定の属性値ではなく、属性の存在のみを検索します。Login Script 属性を持つエントリをすべて検索するクエリは、存在インデックスを使用します。
- ◆ **下位文字列：**属性値文字列のサブセットを照合します。たとえば、「der」という値を含む「LastName」を検索するクエリを実行すると、「Derington」、「Anderson」、および「Lauder」が照合の結果として返されます。

下位文字列インデックスは、作成や維持を行うときに最も多くのリソースが消費されるインデックスです。


8 [OK] をクリックすると、インデックステーブルが更新されます。

9 [適用] をクリックすると、リンバがバックグラウンドプロセスとして再起動され、変更内容が有効になります。




## インデックスを削除する

作成したインデックスが不要になる場合があります。必要のない、ユーザ定義または自動で作成したインデックスは、削除できます。プレディケート統計を使用して、重要度の低いインデックスを調べることができます。詳細については、[168 ページの「プレディケートデータ」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インデックス管理] の順にクリックします。
- 3 利用可能なサーバのリストからサーバを選択します。
- 4 [インデックスの変更] ページで、削除するユーザインデックスまたは自動追加インデックスを選択します。
- 5 [削除] をクリックすると、インデックステーブルが更新されます。
- 6 [適用] をクリックすると、リンバがバックグラウンドプロセスとして再起動され、変更内容が有効になります。

## インデックスをオフラインにする

一時的にインデックスをオフラインにすることで、処理のピーク時にパフォーマンスを調整できます。たとえば、ユーザ定義のインデックスの使用をすべて中断すると、バルクロードを高速化できます。オブジェクトを追加または変更するときは定義されているインデックスを更新する必要があり、すべてのインデックスをアクティブにするとデータのバルクロードの速度が遅くなるためです。バルクロードが完了すると、再びインデックスをオンラインにできます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インデックス管理] の順にクリックします。
- 3 利用可能なサーバのリストからサーバを選択します。
- 4 [インデックスの変更] ページで、オフラインにするインデックスを選択して、[変更状況] をクリックします。

表示されているテーブルでは、インデックスの状態が [オンライン] から [オフライン] に変わります。インデックスは、次のいずれかの状態になります。


- **オンライン**: 現在実行中です。
- **オフライン**: 一時停止中です。[オンライン] をクリックすると再開します。
- **新規**: オンラインに移動するために待機中です。
- **削除済**: インデックステーブルから削除するために待機中です。

- 5 [適用] をクリックします。

## 他のサーバ上でインデックスを管理する

あるサーバで便利に使用されているインデックスがあり、このインデックスを他のサーバでも使用する場合は、他のサーバにインデックス定義をコピーできます。またプレディケートデータを調べると、これとは逆のケースが発生する場合があります。つまり、複数のサーバで使用されていたインデックスが、そのいずれかのサーバで不要になるといったケースです。このような場合、インデックスが不要になったサーバからインデックスを削除できます。

インデックスマネージャを使用すると、他のインスタンスに影響を与えずに、インデックスの1つのインスタンスを処理できます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インデックス管理] の順にクリックします。
- 3 利用可能なサーバのリストからサーバを選択します。
- 4 同じツリーの別のサーバにインデックス定義をコピーするには、[インデックス位置の変更] をクリックします。
- 5 コピーするインデックス定義を選択します。  
インデックスを1つ選択すると、そのインデックスを提供するツリー内のサーバが一覧表示されます。
- 6 このカラムを使用して、インデックスのコピーを目的のサーバに移動します。
- 7 [適用] をクリックします。

## Novell インポート / エクスポート変換ユーティリティを使用してインデックスを管理する

Novell インポート / エクスポート変換ユーティリティを使用してインデックスを作成または削除できます。

インデックスを作成または削除するには、LDIF ファイルを使用する必要があります。LDIF ファイルがインポートされたら、リンバを開始してインデックス処理を初期化することができます。その他の場合には、インデックス処理はリンバが開始されたときに自動的に行われます。

LDIF ファイルにインデックスを指定するには値が必要です。次の場合、ドル (\$) 記号で区切られた文字列は無視されます。

順序	文字列	説明
1	Index version	今後の使用のために予約されています。eDirectory では、常に0に設定します。
2	Index name	ユーザ定義のインデックス名(姓、郵便番号など)を指定します。文字列には、ドル (\$) 記号は使用できません。
3	Index state	インデックスの状態を指定します。インデックスを定義する場合、このフィールドには2(オンライン)を設定します。eDirectory では次の値がサポートされています。 <ul style="list-style-type: none"><li>◆ 0 - 一時停止。該当するインデックスがクエリに使用されず、更新もされていない状態を示します。</li><li>◆ 1 - オンライン開始中。該当するインデックスが作成中であることを示します。</li><li>◆ 2 - オンライン。該当するインデックスが使用されていることを示します。</li><li>◆ 3 - 作成待機中。該当するインデックスの定義が完了し、バックグラウンド処理で実行されるまで待機中であることを示します。</li></ul> バックグラウンドプロセスは、インデックスの構築を開始した後に状態を変更します。

順序	文字列	説明
4	Index rule	<p>マッチングのタイプを指定します。</p> <ul style="list-style-type: none"> <li>◆ 0 - 値一致。値全体または値の最初の部分を含むクエリを最適化します。たとえば、「Jensen」または「Jen」ではじまる文字列と一致する surname 属性を持つエントリのクエリなどがあります。</li> <li>◆ 1 - 存在一致。属性の存在のみを含むクエリを最適化します。たとえば、surname 属性を持つすべてのエントリのクエリなどがあります。</li> <li>◆ 2 - 下位文字列一致。一致する文字列を含むクエリを最適化します。たとえば、「der」の文字列を含む surname 属性を持つエントリのクエリなどがあります。このクエリでは、「Derington」「Anderson」および「Lauder」といった surname 属性を持つエントリが返されます。</li> </ul>
5	Index type	<p>インデックスの作成者を指定します。インデックスを定義する場合、この値には 0 を設定する必要があります。eDirectory では次の値がサポートされています。</p> <ul style="list-style-type: none"> <li>◆ 0 - ユーザ定義</li> <li>◆ 1 - 属性作成時に追加</li> <li>◆ 2 - 操作時に必要</li> <li>◆ 3 - システムインデックス</li> </ul>
6	Index value state	<p>インデックスのソースを指定します。インデックスを定義する場合、この文字列には 1 を設定します。eDirectory では次の値がサポートされています。</p> <ul style="list-style-type: none"> <li>◆ 0 - 未初期化</li> <li>◆ 1 - サーバから追加</li> <li>◆ 2 - ローカル DIB から追加</li> <li>◆ 3 - ローカル DIB から削除</li> <li>◆ 4 - ローカル DIB から変更</li> </ul>
7	Attribute name	<p>属性の NDS 名を指定します。eDirectory では多くの属性が LDAP 名と NDS 名の両方を持っています。この文字列には NDS 名が必要です。</p>

### インデックスを作成する LDIF ファイルの例

```
dn: cn=testServer-NDS,o=Novell
changetype: modify
add: indexDefinition
indexDefinition: 0$indexName$2$2$0$1$attributeName
```

### インデックスを削除する LDIF ファイルの例

```
dn: cn=osg-nw5-7, o=Novell
changetype: modify
```

delete: indexDefinition

indexDefinition: 0\$indexName\$2\$2\$0\$1\$attributeName

## プレディケートデータ

プレディケートデータは、検索の対象となったオブジェクトに関する、サーバ固有の履歴を表します。このデータおよびデータの収集は、eDirectory のインストール時に作成された `ndsPredicateStats` オブジェクトを通して管理されます。`ndsPredicateStats` オブジェクトの名前は、サーバ名に「-PS」が追加されたものになります。

プレディケートデータを使用して最も頻繁に検索されているオブジェクトを識別し、そのオブジェクトのインデックスを作成して今後のアクセスを高速化できます。

## プレディケートデータを管理する

プレディケート統計は、常時実行する機能ではありません。プレディケート統計の収集は、検索のパフォーマンスに影響を与えます。また、データベースのサイズが大きい場合には、統計の収集に長時間かかります。特定のディレクトリの検索で、パフォーマンスの低下が問題になっていると思われる場合に、プレディケート統計を使用します。

ConsoleOne の [プレディケートデータ] プロパティページを使用して、データの収集を管理します。

- 1 ConsoleOne で、サーバオブジェクトを右クリックします。
- 2 [プロパティ] > [プレディケートデータ] > [プロパティ] の順にクリックします。
- 3 `ndsPredicateStats` オブジェクトに対して適切な設定を入力します。

[更新間隔] : データ表示をリフレッシュし、データをディスクに書き込むまでの秒数を指定します。

[詳細] > [有効] : 収集プロセスをバックグラウンドで実行するか、または実行しないかを指定します。データの収集をオフにすると、最後に収集されたデータがメモリから解放されます。[ディスクへ書き込む] を選択すると、データはディスクへ書き込まれます。

[詳細] > [ディスクへ書き込む] : プレディケートデータの格納場所を決定します。常にメモリに格納するか、[更新間隔] で指定したディスクにメモリから書き込むかを指定します。

[詳細] > [値の表示] : データを省略して表示するか、または完全に表示するかを選択します。省略表示では、インデックスの対象に適しているプレディケートを判断するために必要な情報が表示されます。

- 4 [OK] をクリックすると、オブジェクト設定が更新されます。

## eDirectory Service Manager

eDirectory Service Manager では、使用可能な eDirectory サービスおよびそのステータスについての情報が提供されます。また、Service Manager からこれらのサービスを開始または停止できます。

Service Manager は eDirectory サービスのみを管理します。`dsservcfg.xml` 設定ファイルを使用して管理を行います。このファイルには、各プラットフォームで管理できるサービスを表示します。リストにサービスを追加または削除することもできます。

eDirectory Service Manager には次の方法でアクセスできます。

- ◆ [169 ページの「eMBox クライアントのサービスマネージャ eMTool を使用する」](#)
- ◆ [170 ページの「Novell iManager でサービスマネージャプラグインを使用する」](#)

## eMBox クライアントのサービスマネージャ eMTool を使用する

eMBox(eDirectory Management Toolbox)クライアントはコマンドラインで実行される Java クライアントで、eDirectory Service Manager eMTool にリモートアクセスできます。eDirectory の一部としてサーバ上に `emboxclient.jar` ファイルがインストールされます。JVM を実行しているコンピュータならどこからでも実行できます。eMBox クライアントの詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

eMBox クライアントのサービスマネージャ eMTool は、次の手順で使用します。

- 1 非対話式モードで eMBox クライアントを実行するには、コマンドラインに次のコマンドを入力します。

```
java -cp path_to_the_file/emboxclient.jar embox -i
```

(クラスパスにすでに `emboxclient.jar` ファイルが含まれている場合、「`java embox -i`」と入力するだけです)。

次の eMBox クライアントのプロンプトが表示されます。

```
eMBox Client>
```

- 2 Service Manager を実行するサーバにログインするには、次のコマンドを入力します。

```
login -sserver_name_or_IP_address -pport_number  
-uusername.context -wpassword -n
```

ポート番号は通常 80 または 8008 です。ただし、すでにそのポートを使用している Web サーバが存在する場合は異なります。-n オプションを使用すると、セキュリティ保護されていない接続を開始します。

eMBox クライアントはログインが成功したかどうかを表示します。

- 3 次のいずれかの Service Manager コマンドを入力します。

コマンド	説明
<code>service.serviceList</code>	利用できる eDirectory サービスを表示します。
<code>service.serviceStart -nModule_name</code>	指定された eDirectory サービスを開始します。
<code>service.serviceStop -nModule_name</code>	指定された eDirectory サービスを停止します。
<code>service.serviceInfo -nModule_name</code>	指定されたサービスに関する情報を表示します。

eMBox クライアントで `list -tservice` コマンドを使用して、Service Manager オプションの詳細を表示することもできます。詳細については、[483 ページの「eMTool とそのサービスを表示する」](#)を参照してください。


- 4 eMBox クライアントからログアウトするには、次のコマンドを入力します。






```
logout
```

- 5 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

## Novell iManager でサービスマネージャプラグインを使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [サービスマネージャ] の順にクリックします。
- 3 管理するサーバを指定し、[OK] をクリックします。
- 4 選択したサーバの認証を行い、[OK] をクリックします。
- 5 eDirectory サービスの状態をチェックしたり、サービスを開始または停止するには、次のアイコンを使用します。

アイコン	説明
	サービスは実行中です。
	サービスは停止しています。
	サービスを開始します。
	サービスを停止します。
	サービスは実行中ですが、停止できません。

# 7

## Novell iMonitor 2.1 の使用

Novell® iMonitor は、eDirectory™ ツリー内にあるすべてのサーバに対して、複数プラットフォーム対応の監視および診断機能を提供します。Novell iMonitor ユーティリティを使用すると、ネットワーク上で Web ブラウザを使用できる場所ならどこからでもサーバを監視できます。

また iMonitor により、eDirectory 環境に対して、パーティション、レプリカ、またはサーバベースの詳細な管理が可能になります。また、実行しているタスクの種類、タスクの開始時間、結果、および実行時間を検証できます。

iMonitor は、Novell が従来提供していたサーバベースの eDirectory ツール (DSBrowse、DS トレース、DSDiag、および DSRepair が持つ診断機能など) に対して、その代わりに使用できる機能や置き換えることのできる機能を提供します。これらの機能は、Web ベースで利用できます。このため、iMonitor の機能は主にサーバで動作することに重点を置いています。つまり eDirectory ツリー全体ではなく、個々の eDirectory エージェント (ディレクトリサービスで実行しているインスタンス) の状態が、iMonitor の機能に対して重要な要素となります。

iMonitor 2.1 の機能は次のとおりです。

- ◆ eDirectory ヘルスママリ
  - ◆ 同期情報
  - ◆ 認識されているサーバ
  - ◆ エージェントの環境設定
- ◆ eDirectory ヘルスチェック
- ◆ ハイパーリンク付きの DS トレース
- ◆ エージェントの環境設定
- ◆ エージェントアクティビティおよび Verb 統計
- ◆ レポート
- ◆ エージェント情報
- ◆ エラー情報
- ◆ オブジェクト / スキーマブラウザ
- ◆ DirXML® モニタ
- ◆ 検索
- ◆ パーティションリスト
- ◆ エージェントプロセスのステータス
- ◆ バックグラウンドプロセスのスケジュール

- ◆ DSRepair
- ◆ 接続監視

iMonitor の情報は、次の要素に基づいて表示されます。

- ◆ 確立された識別情報

iMonitor で実行するすべての要求は、識別情報に基づく eDirectory 権によって制限されます。たとえば、[DSRepair] ページにアクセスするには、アクセスを行うサーバに対して、サーバの管理者またはコンソールオペレータとしてログインする必要があります。

- ◆ 監視している eDirectory エージェントのバージョン

新しいバージョンの NDS<sup>®</sup> および eDirectory には、以前のバージョンにはない機能とオプションがあります。

iMonitor に表示された情報から、ローカルサーバの状態が一目でわかります。

この章では次のトピックについての情報を説明します。

- ◆ [172 ページの「システム要件」](#)
- ◆ [173 ページの「iMonitor へアクセスする」](#)
- ◆ [173 ページの「iMonitor のアーキテクチャ」](#)
- ◆ [179 ページの「iMonitor の機能」](#)
- ◆ [196 ページの「セキュリティ保護された iMonitor 操作の実現」](#)

## システム要件

iMonitor 2.1 を使用するには次のソフトウェアが必要です。

- ◆ Internet Explorer 5.5 以降、または Netscape 7.02 以降
- ◆ Novell eDirectory 8.7.1 以降

## プラットフォーム

iMonitor 2.1 ユーティリティは次のプラットフォームで動作します。

- ◆ NetWare<sup>®</sup> 5.1 Support Pack 4 以降  
Novell iMonitor は AUTOEXEC.NCF に格納されます。
- ◆ Windows NT、2000、および 2003 Server (SSL なし)
- ◆ Linux
- ◆ Solaris
- ◆ AIX
- ◆ HP-UX

NetWare および Windows では、eDirectory が実行されると、iMonitor は自動的にロードされます。Linux、Solaris、AIX、および HP-UX の場合、ndsmonitor コマンドで -l オプションを使用して iMonitor をロードできます。また、/etc/ndsmonitor.conf ファイルに [ndsmonitor] を追加して、eDirectory サーバを開始する前に iMonitor を自動的にロードすることもできます。



## 監視できる eDirectory のバージョン

iMonitor を使用して監視できる NDS および eDirectory のバージョンは次のとおりです。

- ◆ NetWare 4.11 用以降のすべてのバージョンの NDS および eDirectory
- ◆ Windows 用のすべてのバージョンの NDS および eDirectory
- ◆ Linux、Solaris、AIX、および HP-UX 用のすべてのバージョンの NDS および eDirectory

## iMonitor へアクセスする

- 1 iMonitor の実行ファイルが eDirectory サーバで実行されていることを確認します。
- 2 Web ブラウザを開きます。
- 3 アドレス (URL) のフィールドに、次の形式で入力します。

**`http://server's_TCPIP_address:httpstack_port/nds`**

たとえば、次のように入力します。

`http://137.65.135.150:8008/nds`

DNS 名は、iMonitor 内でサーバの IP または IPX™ アドレス、もしくは識別名を使用できる箇所であれば常に使用できます。たとえば、次のような DNS が設定されているとします。

`http://prv-gromit.provo.novell.com/nds?server=prv-igloo.provo.novell.com`

これは、次の設定と同等です。

`http://prv-gromit.provo.novell.com/nds?server=IP_or_IPX_address`

または

`http://prv-gromit.provo.novell.com/nds?server=/cn=prv-igloo,ou=ds,ou=dev,o=novell,t=novell_inc`

eDir HTTPS スタックが有効であれば、HTTPS を通して iMonitor を使用できます。

- 4 ユーザ名、コンテキスト、パスワードを指定します。  
すべての機能にアクセスするには、完全識別名を指定して管理者としてログインするか、管理者と同等のアクセス権でログインします。
- 5 [ログイン] をクリックします。

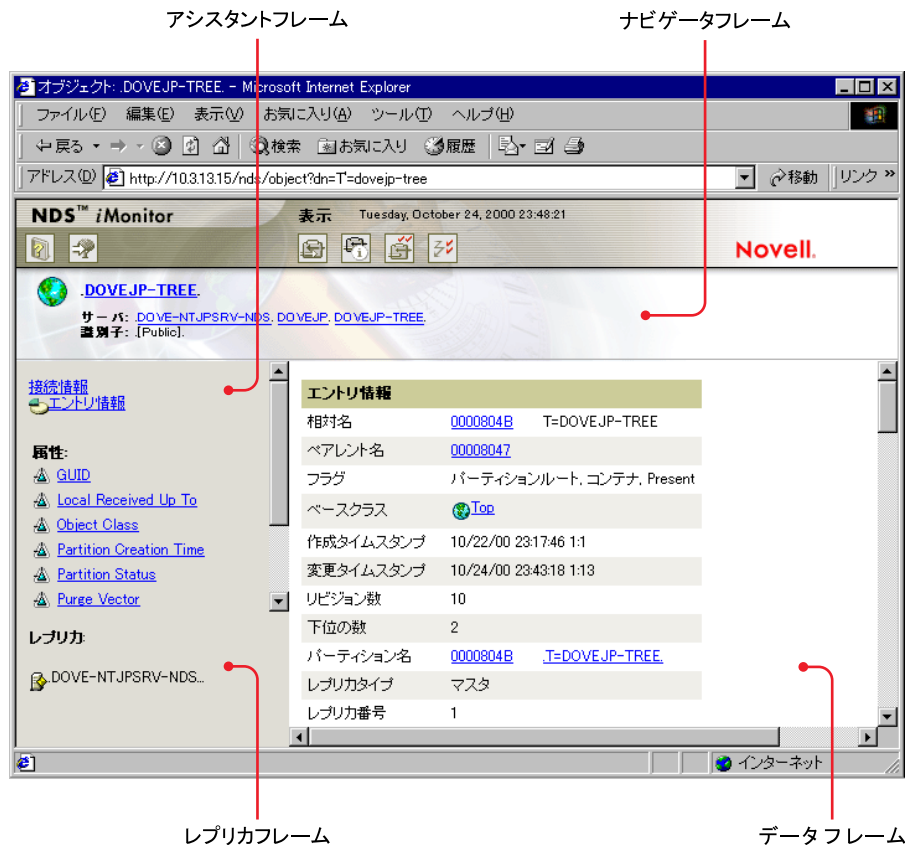
## iMonitor のアーキテクチャ

- ◆ 174 ページの「iMonitor ページの構成」
- ◆ 175 ページの「動作モード」
- ◆ 176 ページの「すべてのページからアクセス可能な iMonitor の機能」
- ◆ 176 ページの「NetWare Remote Manager との統合」
- ◆ 177 ページの「環境設定ファイル」

## iMonitor ページの構成

iMonitor の各ページは、ナビゲータフレーム、アシスタントフレーム、データフレーム、およびレプリカフレームの4つのフレームまたはセクションに分かれています。

図 26 iMonitor の各フレーム



**ナビゲータフレーム:** ページの上部にあります。このフレームには、データの読み込み元のサーバ名、ユーザの識別情報、および他の画面 (オンラインヘルプ、ログイン、サーバポータルなどの iMonitor ページ) にリンクするためのアイコンが表示されます。

**アシスタントフレーム:** ページの左側にあります。このフレームには、ナビゲーション用の項目 (他のページへのリンクなど)、データフレームでのデータの検索に使用する項目、および表示されているページでのデータの取得や解釈に使用する項目が含まれます。

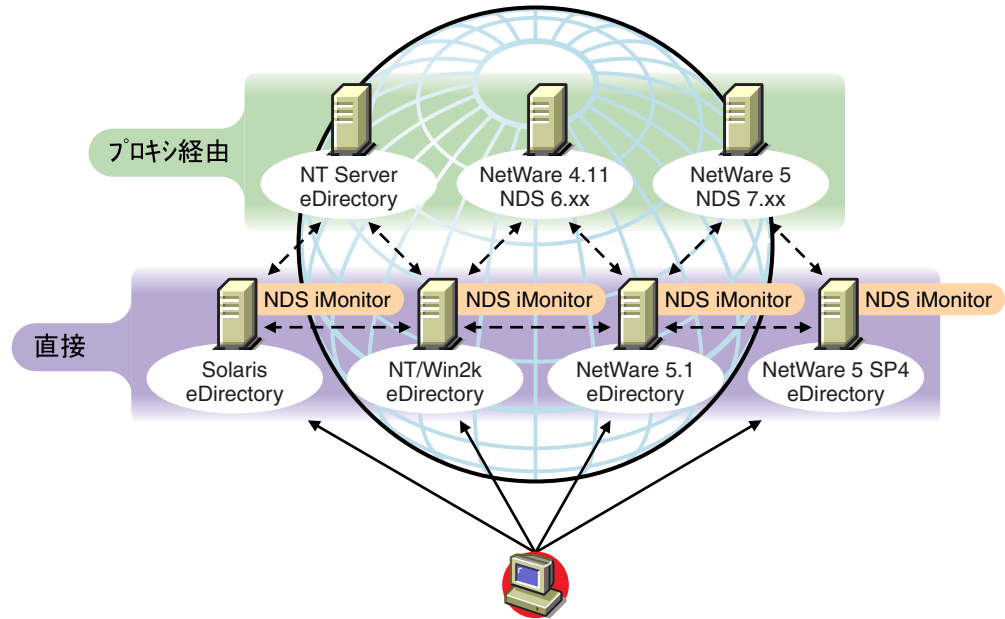
**データフレーム:** 上部にあるリンクをクリックすると、ローカルサーバに関する詳細情報が表示されます。Web ブラウザがフレームをサポートしていない場合には、このページだけが表示されます。

**レプリカフレーム:** 現在表示されているレプリカを判別できます。またリンクを使用して、現在表示されている情報が、他のサーバやレプリカを基準としたときに、どのような状態になっているかを確認できます。表示したページに、要求したデータの他のレプリカが存在する場合、またはデータフレームに表示されている情報を別の状態で表示するレプリカが存在する可能性がある場合にのみ、レプリカフレームが表示されます。

## 動作モード

Novell iMonitor には、ダイレクトモードとプロキシモードという 2 種類の動作モードがあります。モードを切り替えるために環境設定情報を変更する必要はありません。モードは Novell iMonitor が自動的に切り替えますが、eDirectory ツリーのナビゲートを正しく効率的に行うために、これらのモードについて理解しておくことをお勧めします。

図 27 動作モード



**ダイレクトモード:**ダイレクトモードが使用されるのは、Web ブラウザが iMonitor の実行ファイルを実行しているコンピュータ上のアドレスまたは DNS 名を直接ポイントしていて、そのコンピュータのローカル eDirectory DIB 上の情報だけを読み込んでいる場合です。

iMonitor の機能の中にはサーバ限定のものもあります。ここでいう「サーバ限定」の機能とは、ローカルコンピュータ上で動作している iMonitor 以外からは使用できない機能のことです。サーバ限定の機能では、リモートからアクセスできないローカル API のセットを使用します。サーバ限定の iMonitor の機能には、DS トレース、DSRepair、および [バックグラウンドプロセスのスケジュール] ページなどがあります。ダイレクトモードの場合、すべての iMonitor 機能がローカルコンピュータから利用できます。

ダイレクトモードの主な特徴は次のとおりです。

- ◆ サーバ限定の機能をすべて使用できます。
- ◆ ネットワークの通信量が減少します (高速アクセスが可能)。
- ◆ eDirectory のバージョンに関係なく、プロキシによるアクセスも可能です。

**プロキシモード:**プロキシモードが使用されるのは、Web ブラウザが、1 つのコンピュータ上で実行されている iMonitor をポイントしていて、同時に他のコンピュータから情報を読み込んでいる場合です。iMonitor では、サーバ限定でない機能に対してはサーバ限定でない従来の eDirectory プロトコルを使用するため、NDS 6.x 以降の従来のバージョンの eDirectory でも監視や診断の対象にできます。ただし、サーバ限定の機能では、リモートからアクセスできない API が使用されます。

プロキシモードが有効なときに、他のサーバの動作モードをダイレクトモードに切り替えることもできます。ただし、そのサーバの eDirectory のバージョンで iMonitor がサポートされていることが条件です。プロキシによる情報収集対象のサーバ上で iMonitor が実行されている場合は、ナビゲータフレームに追加のアイコンボタンが表示されます。カーソルをこのアイコン上に移動すると、そのリモートサーバ上で実行されているリモート iMonitor へのリンクが表示されます。ただし、情報収集対象のリモートサーバで以前のバージョンの eDirectory が実行されている場合は、このアイコンは表示されません。そのリモートサーバが iMonitor をサポートしているバージョンの eDirectory にアップグレードされるまでは、そのサーバからの情報収集には常にプロキシを使用する必要があります。

プロキシモードの主な特徴は次のとおりです。

- ◆ ツリー内のすべてのサーバで iMonitor を実行しなくても、iMonitor の機能の大部分を利用できます
- ◆ 1 つのサーバをアップグレードするだけで済みます
- ◆ 1 つのアクセスポイントでダイヤルインが可能です
- ◆ iMonitor 自体へのアクセスには低速なリンクを使用し、iMonitor から eDirectory 情報へのアクセスには高速なリンクを使用できます
- ◆ 以前のバージョンの NDS の情報にアクセスできます
- ◆ サーバ限定の機能は、iMonitor がインストールされているコンピュータ以外では使用できません

## すべてのページからアクセス可能な iMonitor の機能

エージェントの要約、エージェント情報、エージェントの環境設定、トレースの環境設定、DSRepair、レポート、および検索の各ページには、ナビゲータフレームを使用することによってどの iMonitor ページからでもリンクできます。その他、どの iMonitor ページからでも、Novell Support Web ページにログインおよびリンクできます。

**[ログイン] ボタンと [ログアウト] ボタン :** ログインしていない状態では、[ログイン] ボタンが有効になります。システムにログインしている状態では [ログアウト] ボタンが表示され、これを使用するとブラウザウィンドウを閉じることができます。ブラウザウィンドウがすべて閉じられるまでは iMonitor セッションは開いたままになるため、再びログインする必要はありません。自分のログインステータスは、ナビゲータフレームに表示された識別情報を調べることによって、どのページからでも確認できます。

**Support Connection ページへのリンク :** ページ右上に表示される Novell のロゴは、Novell Support Connection Web ページへのリンクとして使用できます。ここから Novell の Web サイトに直接リンクして、最新のサーバパッチキット、更新データ、各製品に固有なサポート情報などを取得できます。

## NetWare Remote Manager との統合

NetWare 5 以降のサーバからは NetWare Remote Manager へリンクでき、NetWare サーバの監視、診断、およびトラブルシューティング情報の取得が Web ベースで実行できます。

iMonitor と NetWare Remote Manager を統合する方法を次に示します。

- ◆ NetWare Remote Manager のライトウェイト Web サーバ (<https://nwm.novell.com>) は、NetWare プラットフォームに対して、iMonitor アーキテクチャの第 1 層を割り当てます。

- ◆ iMonitor が NetWare Remote Manager (portal.nlm) に登録されます。これで、iMonitor および他の eDirectory 固有の情報へのリンクが、NetWare Remote Manager インタフェース経由で使用可能になります。

これらのリンクは、Remote Manager インタフェースの [eDirectory の管理] セクションに表示されます。また eDirectory エージェントのヘルス情報へのリンクが、eDirectory 関連のカテゴリ内にあるヘルスマニタの [Diagnose Server(サーバの診断)] セクションに表示されます。

また、NetWare Remote Manager が eDirectory に登録されます。これにより iMonitor と NetWare Remote Manager 間のクロスリファレンスが可能になり、各ツール間の移動が、よりスムーズになります。

## 環境設定ファイル

iMonitor に含まれる環境設定ファイルを使用すると、ユーティリティのデフォルトの動作や値を変更したり、設定することができます。

環境設定ファイルはテキストファイルで、必要な値が指定された環境設定パラメータタグが含まれています。このファイルは NetWare および Windows 上では iMonitor の実行可能ファイルと同じディレクトリ (通常 Novell eDirectory の実行可能ファイルと同じ場所)、または Linux、Solaris、AIX、および HP-UX 上では /etc ディレクトリにあります。

- ◆ [177 ページの「ndsimon」](#)
- ◆ [178 ページの「ndsimonhealth」](#)

### ndsimon

ndsimon の環境設定ファイルでは、トレースファイルの設定の変更、サーバへのアクセス制御、コンテナのリスト表示または検索結果を表示する際のオブジェクトの最大表示数の設定、およびアイドル状態が何分続くと接続がログアウトするかを指定できます。

サーバ	環境設定ファイル
NetWare	sys:\system\ndsimon.ini
Windows NT および Windows 2000	install\directory\novell\NDS\ndsimon.ini
Linux、Solaris、AIX、HP-UX	/etc/ndsimon.conf

ndsimon の環境設定ファイルに設定するパラメータには、次のような 2 種類のグループがあります。

- ◆ iMonitor の実行可能ファイル自体の実行方法に適用されるパラメータ

NetWare の場合を除き、iMonitor の実行可能ファイルはロードされると、従来の HTTP ポート 80 でリッスンします。このポートが使用中の場合、待機するポートを 8008 に切り替えます。ポート 8008 が使用中の場合は、iMonitor はさらにポートを切り替え、番号を 2 ずつ増やしながら (8010、8012 など) 8078 に達するまで使用可能なポート番号を検索します。

SSL が設定され使用可能になっている場合も、同様のパターンでバインドが実行されます。この場合、最初にポート 81 がバインドされ、次に 8009、8011、8013 と続きます。

これにより、iMonitor と、同じサーバで実行している Web サーバとの共存が可能になります。プラットフォームによっては、インストールされた Web サーバをロード

する前に iMonitor をロードできます。また、iMonitor をバインドするポートを選択することもできます。通常のポートおよび SSL ポートは、HttpPort および HttpsPort パラメータをそれぞれに使用して設定できます。付属の環境設定ファイルに、サンプルがコメントアウトされて含まれています。デフォルトでは、iMonitor は、ロードするサーバのすべての NIC アドレスにバインドされます。ただし、[アドレス] パラメータを使用して、バインドするアドレスをコンマ区切り形式のリストで指定できます。

NetWare でも、ポート選択について同様の規則が使用されますが、この規則は NetWare Remote Manager HTTP スタック (httpstk.nlm) によって制御されます。この規則の詳細は、NetWare Remote Manager のマニュアルに記載されています。

- ◆ 特定の機能またはページに適用されるパラメータ

iMonitor に付属する環境設定ファイルには、変更可能なパラメータのサンプルが含まれています。これらのパラメータの先頭にはシャープ記号 (#) が付いています。これは、パラメータがコメントアウトされていることを示していて、iMonitor が環境設定ファイルを解析するときには、これらのパラメータは無視されます。付属する環境設定ファイルでは、これらのパラメータにはすべて、内部でバインドされたデフォルト値が使用されます。これらのパラメータを使用可能にする、またはパラメータを追加するには、行の先頭の「#」を削除します。

## ndsimonhealth

ndsimonhealth の環境設定ファイルでは、[エージェントヘルス] ページのデフォルト設定を変更できます。[エージェントヘルス] オプションを有効または無効にしたり、オプションのレポートレベルおよび範囲を設定したり、サーバのレポートレベルを設定できます。

サーバ	環境設定ファイル
NetWare	sys:\system\ndsimonhealth.ini
Windows NT および Windows 2000	install directory\novell\NDS\ndsimonhealth.ini
Linux、Solaris、AIX、HP-UX	/etc/ndsimonhealth.conf

ndsimonhealth の環境設定ファイルに設定するオプションには、次のような 3 種類のオプションがあります。

- ◆ 有効または無効に設定するだけのオプション

オプションを無効にするには、オプションの前のシャープ記号 (#) を削除し、コロン (:) の後ろにリスト表示されるすべてのレベルを「OFF」に置き換えます。これらのオプションのレポートレベルを設定するには、オプションの前の「#」文字を削除し、コロンの後ろにレポートレベルを追加します。有効なレベルは、WARN、MARGINAL、および SUSPECT です。これらのオプションに入力できるレポートレベルは 1 つだけです。

- ◆ 設定の範囲を指定する一般オプション

これらのオプションでは、レポートレベルの設定を有効または無効にしたり、レポートレベルを設定したりできます。またレポートレベルの範囲の設定もできます。

これらのすべてのオプションのレポートレベルを設定するには、オプション名の後に -active と記述し、その後ろに設定するレポートレベルを記述します。たとえば、time\_delta をアクティブに設定するには、環境設定ファイルに次の行を追加します。

```
time_delta-active:WARN
```

`time_delta` を非アクティブに設定するには、環境設定ファイルに次の行を追加します。

```
time_delta-active:OFF
```

範囲を入力する場合、指定する範囲はこのレポートレベルを表示しない範囲です。

3 つすべてのレポートレベルをアクティブにするオプションの設定方法、および範囲の設定方法については、次の `time_delta` の例を参照してください。この例では、-2 ~ 2 の範囲外では少なくとも `marginal` のレベルが表示され、-5 ~ 5 の範囲外では少なくとも `suspect` のレベルが表示され、-10 ~ 10 の範囲外では `warning` のレベルが表示されます。

```
time_delta-active:WARN | SUSPECT | MARGINAL
time_delta-Min_Warn:      -10
time_delta-Min_Suspect:   -5
time_delta-Min_Marginal:  -2
time_delta-Max_Marginal:   2
time_delta-Max_Suspect:   5
time_delta-Max_Warn:     10
```

これらのオプションのヘルプを表示するには、iMonitor で次の URL を入力します。

```
http://XXX.XXX.XXX.XXX:PORT/nds/help?hbase=/nds/health/OPTION_NAME
```

`XXX.XXX.XXX.XXX:PORT` には iMonitor がアクセスできる IP アドレスとポート、`OPTION_NAME` にはヘルプ表示するオプション名 (`time_delta` など) を入力します。

現在の設定レベルと範囲を表示するには、ブラウザを使用して表示するオプションを含むヘルスページへ進み、ブラウザの URL 行の最後に次を追加します。

```
&op=setup
```

- ◆ カスタム設定または複合設定が必要なオプション

設定できるレポートレベルには次の 3 種類があります。

- ◆ **WARN** は、すぐにアップグレードする必要があるバージョンの eDirectory を実行しているサーバを検出します。
- ◆ **SUSPECT** は、アップグレードが望まれるバージョンの eDirectory を実行しているサーバを検出します。
- ◆ **MARGINAL** は、最新バージョンではない eDirectory を実行しているサーバを検出します。

これらのオプションは、サーバのバージョンが指定された許容範囲にあるかどうかのレポートレベルを設定します。

## iMonitor の機能

このセクションでは iMonitor の機能について簡単に説明します。


iMonitor が持つ各機能の詳細については、オンラインヘルプの該当するセクションを参照してください。

- ◆ 180 ページの「eDirectory サーバのヘルス情報の表示」
- ◆ 181 ページの「パーティション同期ステータスの表示」
- ◆ 181 ページの「サーバ接続情報の表示」
- ◆ 182 ページの「認識されているサーバの表示」

- ◆ 182 ページの「レプリカ情報の表示」
- ◆ 183 ページの「DS エージェントを制御および環境設定する」
- ◆ 184 ページの「トレースを環境設定する」
- ◆ 185 ページの「プロセスステータス情報の表示」
- ◆ 185 ページの「エージェントアクティビティの表示」
- ◆ 186 ページの「トラフィックパターンの表示」
- ◆ 186 ページの「バックグラウンドプロセスの表示」
- ◆ 186 ページの「eDirectory サーバエラーの表示」
- ◆ 187 ページの「DSRepair 情報の表示」
- ◆ 187 ページの「エージェントのヘルス情報の表示」
- ◆ 187 ページの「ツリー内のオブジェクトの参照」
- ◆ 188 ページの「同期またはパージのためのエントリの表示」
- ◆ 189 ページの「レプリカの同期ステータスの表示」
- ◆ 189 ページの「レポートの設定と表示」
- ◆ 190 ページの「スキーマ、クラス、および属性定義の表示」
- ◆ 191 ページの「オブジェクトの検索」
- ◆ 191 ページの「ストリームビューアの使用」
- ◆ 192 ページの「DIB セットのクローン」

## eDirectory サーバのヘルス情報の表示

[エージェントの要約] ページでは、同期設定、エージェントプロセスのステータス、データベースで認識されているサーバの総数など、eDirectory サーバのヘルス情報を表示できます。

**1** iMonitor で、[エージェントの要約]  をクリックします。

**2** 次のオプションから選択します。

**[エージェント同期の概要]** レプリカの数とタイプ、およびこれらレプリカが正常に同期されてから経過した時間を表示できます。その他、レプリカのタイプ別にエラーの数を表示することもできます。表示できるレプリカまたはパーティションが 1 つだけの場合、項目名は「パーティション同期ステータス」になります。

[エージェント同期の概要] が表示されない場合は、ユーザの識別情報に基づく権利で表示できるレプリカがないことを意味します。

**[データベースで認識されているサーバ合計]** ローカルデータベースが認識しているサーバのタイプと数、および各サーバが実行中であるかどうかを表示できます。

**[エージェントプロセスのステータス合計]** エージェント上で実行されているプロセスのステータスを、管理者に依頼せずに自分で調べることができます。ステータス情報は問題や重要な情報が発生したときに記録されます。表示される表のサイズは、記録されているステータスの数によって異なります。



## パーティション同期ステータスの表示

[エージェント同期] ページでは、パーティションの同期状態を表示できます。ページの左側のアシスタントフレームに一覧表示されているオプションから選択して、表示する情報を絞り込むこともできます。

- 1 iMonitor で、アシスタントフレームの [エージェント同期] をクリックします。
- 2 次のオプションから選択します。

[パーティション同期ステータス] パーティション、エラーの数、最終同期時刻、最大リングデルタを表示できます。

[パーティション] パーティションごとの「レプリカ同期」ページへのリンクを表示できます。

[最終同期日時] サーバから個々のパーティションのレプリカをすべて同期できたときから経過した時間を表示できます。

[最大リングデルタ] リング内にあるすべてのレプリカに対して同期できない可能性があるデータ量を表示します。たとえば、ユーザが自分のログインスクリプトを変更してから 30 分が経過していない場合、最大リングデルタへの割り当てが 45 分であれば、このユーザのログインは正常に同期されないおそれがあります。その場合、このユーザはログイン時に前のログインスクリプトを受け取ることになります。ユーザが 45 分以上前にログインスクリプトを変更している場合は、このユーザはすべてのレプリカから常に新しいログインスクリプトを受け取るようになります。

[最大リングデルタ] に [不明] が表示されている場合は、遷移同期ベクトルに不整合があり、レプリカ / パーティション操作が実行中などの理由で最大リングデルタを計算できないことを意味します。

## サーバ接続情報の表示

[エージェント情報] ページでは、ローカルサーバの接続情報を表示できます。

- 1 iMonitor で、アシスタントフレームの [エージェント情報] をクリックします。
- 2 次のオプションから選択します。

[Ping 情報] サーバからのアドバタイズ先アドレスのセットに対して、iMonitor が IP Ping を送信したことを表示します。表示されるのは、応答があった場合です。

[DNS 名] iMonitor がサーバによってサポートされている IP アドレスに対してアドレスの反転を試みたことを表示します。対応する DNS 名が表示されます。

使用しているトランスポート、環境設定、およびプラットフォームによっては、この情報が表示されない場合もあります。

[接続情報] サーバの参照、タイムデルタ、一番ルート側のマスタレプリカ、およびレプリカ深さなどのサーバの情報を表示できます。

使用しているトランスポート、環境設定、およびプラットフォームによっては、この情報が表示されない場合もあります。

[サーバ照会先] ローカルサーバへのアクセスに使用できるアドレスを一括して表示できます。

[時刻同期済み] レプリカの最終タイムスタンプが現在の時刻より大きい値でない限り、合成時刻や将来の時刻は使用されていないことを示します。

eDirectory では、サーバの現在時刻に基づいてタイムスタンプが正しく発行されるよう、時刻の同期が行われることが前提とされています。ただし、時刻同期プロトコルが同期状態にあることは保証されていません。

[**タイムデルタ**] iMonitor とリモートサーバの時刻の差を秒単位で表示できます。負の整数が表示された場合、iMonitor の時刻はリモートサーバの時刻より進んでいます。正の整数が表示された場合は、iMonitor の時刻はリモートサーバの時刻より遅れています。

[**最もルートに近いマスタレプリカ**] 最上位のレプリカつまりネーミングツリーのルートに最も近いレプリカがマスタレプリカであることを示します。

[**レプリカ深さ**] 最上位レプリカの深さ (最上位レプリカとツリーのルートの間レベル数) を表示します。

## 認識されているサーバの表示

[認識されているサーバのリスト] には、ソースサーバのデータベースが認識しているすべてのサーバのリストが表示されます。このリストには、データベースで認識されているすべてのサーバを表示することも、レプリカリング内のすべてのサーバを表示することもできます。サーバの横にアイコンが表示された場合、そのサーバはレプリカリングのメンバーです。

**1** iMonitor で、アシスタントフレームの [認識サーバ] をクリックします。

**2** 次のオプションから選択します。

[**エントリ ID**] ローカルサーバのオブジェクト識別子を表示します。エントリ ID は、複数のサーバ間で共用することはできません。

[**NDS リビジョン**] 通信相手のサーバにキャッシュまたは保存されている eDirectory ビルド番号または NDS バージョンを表示します。

[**ステータス**] サーバのステータス (稼働中、停止中、不明) を表示します。通信相手のサーバのステータスが不明となっている場合、過去にこのサーバと通信する必要がなかったことを表します。

[**最終更新時刻**] 該当するサーバが相手サーバと最後に通信を試みて、その相手サーバが停止中であることを検出したときの時刻を表示します。このカラムが表示されない場合は、すべてのサーバが稼働中であることを意味しています。

## レプリカ情報の表示

[**パーティション**] ページでは、通信相手のサーバ上にあるレプリカに関する情報を表示できます。ページの左側のアシスタントフレームに一覧表示されているオプションから選択して、表示するページの情報を絞り込むことができます。

[**サーバパーティション情報**] 該当するサーバのパーティションについての情報 (エントリ ID、レプリカの状態、ページ時間、最終更新時刻など) を表示できます。

[**パーティション**] サーバのパーティションの Tree オブジェクトに関する情報を表示できます。


[**ページ時間**] すべてのレプリカが削除を認識しているためにすでに削除されたデータを、データベースから削除できる時間を示します。

[**最終変更時刻**] このレプリカのデータベースに書き込まれたデータの、最後に発行されたタイムスタンプを表示できます。これにより、将来の時刻が設定されていないかどうか、および合成時刻が使用されていないかどうかを確認できます。

[**レプリカ同期**] パーティションに対応する [レプリカ同期サマリ] ページを表示できます。[レプリカ同期] ページには、パーティション同期ステータスとレプリカステータスについての情報が表示されます。また、パーティションとレプリカのリストを表示することもできます。

## DS エージェントを制御および環境設定する

[エージェント環境設定] ページでは、DS エージェントの制御および環境設定ができます。このページで利用できる機能は、現在の識別情報に基づく権利および使用している eDirectory のバージョンによって異なります。

- 1 iMonitor で、[エージェント環境設定]  をクリックします。
- 2 次のオプションから選択します。

[エージェント情報] ローカルサーバの接続情報を表示できます。

[パーティション] 通信しているサーバ上にあるレプリカを表示できます。

[レプリケーションフィルタ] 指定した eDirectory エージェントに対して設定されたレプリケーションフィルタを表示できます。NDS eDirectory 8.5(ビルドバージョン 85.xx) は、フィルタ済みレプリカと呼ばれる機能を初めて実装した eDirectory バージョンです。フィルタ済みレプリカの使用と設定方法の詳細については、[125 ページの「フィルタ済みレプリカを設定し管理する」](#)を参照してください。

[エージェントトリガ] バックグラウンドプロセスを開始します。エージェントトリガは、機能的には SET DSTRACE=*option* コマンドと同じです。

[バックグラウンドプロセス処理設定] 特定のバックグラウンドプロセスを実行する時間間隔を変更します。バックグラウンドプロセスの設定は、機能的には SET DSTRACE=*option* コマンドと同じです。

[エージェント同期] インバウンド同期やアウトバウンド同期を無効または有効にします。同期を無効にする期間(単位は時間)を指定できます。

[データベースキャッシュ] DS データベースエンジンが使用するデータベースキャッシュのサイズを設定します。提供されるさまざまなキャッシュ統計情報により、適切な量のキャッシュが利用可能であるかを判断できます。十分なキャッシュがないと、システムのパフォーマンスが悪化する原因となります。

[ログイン設定] ログイン更新のキュー登録を無効にします。更新が有効な場合、更新の時間間隔を増減することもできます。


eDirectory の最新バージョンでは、ログイン速度を向上するパフォーマンス強化機能が実装されています。NDS の以前のバージョンでは、ログイン時に変更を実行する必要があったためユーザの待機時間が発生していましたが、今回の強化により、変更がキューに登録されるようになりました。eDirectory データベースに変更を加えるときは必ずデータベースのロックが必要です。そのため使用率の高い時間帯には、データベースのロックを必要とするリクエストの数がその時点でいくつあるかによって、ログイン時間が非常に長くなったり予測できない場合があります。ロックの必要性をなくし、ログイン更新をキュー登録することにより、ログイン速度が大幅に向上し、予測しやすくなりました。

このオプションでは、異なる eDirectory 環境でキュー登録の動作を制御できます。環境によっては、キュー登録されたデータが重要であれば、即時にデータベースに書き込む必要があります。その場合ユーザは、更新が実行される間待機する必要があります。また別の環境では、データは全く使用されず、無視することができます。デフォルトの動作は、多くの環境に適した設定になっています。

## トレースを環境設定する

[トレースの環境設定] ページでは、トレースを設定できます。Novell iMonitor の DS トレース機能はサーバ限定の機能です。つまり、この機能は iMonitor が動作しているサーバ以外からは起動できません。他のサーバで実行されている DS トレース機能にアクセスするには、そのサーバで実行されている iMonitor に切り替える必要があります。

[トレースの環境設定] ページの情報にアクセスするには、サーバの管理者と同等の権利またはコンソールオペレータの権利が必要です。このページの情報にアクセスする前に、認証情報の確認のためにユーザ名とパスワードを入力する必要があります。

**1** iMonitor で、[トレースの環境設定]  をクリックします。

**2** 次のオプションから選択します。

**[更新]** トレースオプションおよびトレース行プリフィックスに変更を送信できません。DS トレースがオフになっているときに [オン] ボタンをクリックすると、DS トレースがオンになります。DS トレースがすでにオンになっているときに [更新] ボタンをクリックすると、変更内容が現在のトレースに反映されます。

**[オン] / [オフ]** DS トレースをオンまたはオフにします。ボタン上に表示される文字列は、DS トレースが現在オンかオフかによって異なります。DS トレースがオンになっている場合、ボタン上には [トレースオフ] という文字列が表示されます。DS トレースのオンとオフを切り替えるには、このボタンをクリックします。DS トレースがオフになっているときに [トレースオン] をクリックした場合の動作は、[更新] ボタンをクリックした場合と同じです。

**[トレース行プリフィックス]** このオプションを使用すると、トレース行の先頭に追加するデータを選択することができます。


**[DS トレースオプション]** トレースを開始したローカル DS エージェント上のイベントに対して適用されます。[DS トレースオプション] を使用すると、エラーと潜在的な問題、およびローカルサーバ上の eDirectory に関する情報を表示できます。[DS トレースオプション] をオンにすると、CPU の負荷が増え、システムのパフォーマンスが低下します。このため、DS トレースオプションは診断のみに使用して、通常はオフにします。[DS トレースオプション] は、機能的には SET DSTRACE=*+option* コマンドと同じですが、使い易さの点では勝っています。

**[イベント環境設定]** DS トレースで監視のために有効または無効にする eDirectory イベントオプションをリスト表示します。イベントシステムは、オブジェクトの追加、削除、属性値の変更など、ローカルな操作に対してイベントを生成します。個々のイベントタイプに対して、そのイベントタイプに固有の情報が含まれた構造が返されます。

**[トレース履歴]** 以前に実行したトレースのリストを表示できます。各トレースログは、トレースデータの収集期間によって識別されます。

**[トレーストリガ]** DS トレース内で、指定した DS エージェント情報を表示するために設定する必要があるフラグを表示できます。トレーストリガを有効にした場合、トレースに書き込まれる情報の量が多くなります。Novell サポートからの指示があった場合にのみ、トレーストリガを有効にすることをお勧めします。

**3** [トレースオン] をクリックして DS トレースをオンにし、変更を送信します。

**4** iMonitor で DS トレースを表示するには、 または [トレースライブ] をクリックします。

## プロセスステータス情報の表示

[エージェントプロセスのステータス] ページでは、バックグラウンドプロセスのステータスに関するエラーと発生した各エラーの詳細情報を表示できます。ページの左側のアシスタントフレームに一覧表示されているオプションから選択して、このページに表示する情報を絞り込むことができます。

- 1 iMonitor で、アシスタントフレームの [エージェントプロセスのステータス] をクリックします。

現在、バックグラウンドプロセスのステータスで報告される情報には、次のものが含まれます。

- ◆ スキーマの同期
- ◆ 破損通知処理
- ◆ 外部参照 /DRL
- ◆ リンバ
- ◆ 修復

## エージェントアクティビティの表示

[エージェントアクティビティ] ページでは、トラフィックパターンや考えられるシステムボトルネックを調べることができます。このページには、eDirectory で現在処理されているバンプおよび要求を表示できます。また、これらのうち、データベースへの書き込みのために DIB ロックを取得しようとしている要求がどれかを特定したり、DIB ロックの取得待機中の要求数を調べることもできます。

Novell eDirectory 8.6 以降を実行しているサーバを表示すると、パーティションのリスト、およびナビゲータフレームで指定したサーバがあるレプリカリングを対象としているサーバも表示されます。Novell eDirectory 8.6 を導入すると、同期処理はシングルスレッドではなくなります。1つの 8.6 サーバに、1つ以上のレプリケーションパートナーへのアウトバウンドパーティションが同時に複数存在する可能性があります。このため、このような並行同期処理の監視がさらに容易になるよう、[同期アクティビティ] ページが作成されています。

- 1 iMonitor で、アシスタントフレームの [エージェントアクティビティ] をクリックします。
- 2 次のオプションから選択します。

**[Verb Activity and Statistics(バンプアクティビティおよび統計情報)]** eDirectory の最後の初期化以降に呼び出されたバンプの総数や発行されたリクエストの数をリアルタイムで表示できます。また、現在アクティブになっている要求の数と、これらの要求を処理するための最小、最大、および平均時間 (ミリ秒単位) も表示されます。

**[Synchronization Current and Schedule(現在同期およびスケジュール)]** インバウンド同期およびアウトバウンド同期が発生したさまざまな時刻のリストを表示します。インバウンド同期またはアウトバウンド同期が現在実行中の場合、プロセスが実行中であること、そのサイクルが開始された時刻、およびそのプロセスを実行しているサーバを示すアイコンも表示されます。

インバウンド同期およびアウトバウンド同期が無効になっている場合は、現在同期が無効であることおよび再び有効になる予定の時刻を示すアイコンが表示されます。アウトバウンド同期では、次に再び有効になる予定の時刻も表示されます。

[イベント] 現在アクティブな状態にあるイベント、イベントハンドラの統計情報とイベント統計情報サマリ、および呼び出された現在のイベント権利機能のリストを表示できます。

[バックグラウンド処理スケジュール] スケジュールされているバックグラウンド処理、その現在の状態、および再実行のスケジュールを表示できます。

## トラフィックパターンの表示

[Verb 統計] ページでは、トラフィックパターンや考えられるシステムボトルネックを調べることができます。このページでは、eDirectory の最後の初期化以降に呼び出されたバートの総数や発行されたリクエストの数をリアルタイムで表示することもできます。その他、これらリクエストのうち現在アクティブなもの数や、これらリクエストの処理にかかる時間の最大値、平均値、最小値 (それぞれミリ秒単位) も表示できます。バックグラウンドプロセス、バインダリ、および標準 eDirectory リクエストが追跡されます。

このページを以前のバージョンの eDirectory で表示すると、eDirectory 8.5 以降で表示する場合より情報量が少なくなります。

## バックグラウンドプロセスの表示

[バックグラウンド処理スケジュール] ページでは、スケジュールされているバックグラウンドプロセスを、現在の状態と次回の実行予定時刻とともに表示できます。Novell iMonitor のバックグラウンド処理スケジュール機能は、サーバ限定の機能です。つまり、この機能は iMonitor が動作しているサーバ以外からは表示できません。他のサーバで実行されているバックグラウンド処理スケジュール機能にアクセスするには、そのサーバで実行されている iMonitor に切り替える必要があります。eDirectory 8.5 にアップグレードしたサーバの数が増えれば、iMonitor のサーバ限定機能の利用範囲は広がります。その他のサーバ限定機能には、[DS トレース] ページおよび [DSRepair] ページなどがあります。

[バックグラウンド処理スケジュール] ページの情報にアクセスするには、サーバの管理者と同等の権利またはコンソールオペレータの権利が必要です。このページの情報にアクセスするには、認証情報の確認のためにログインする必要があります。

## eDirectory サーバエラーの表示

[エラー索引] ページでは、eDirectory サーバ上で検出されたエラーについての情報を表示できます。検出されたエラーは、eDirectory 固有のエラー、および関連のあるその他のエラーの 2 つのフィールドに分けて表示されます。表示されるエラーのそれぞれに説明がハイパーリンクされていて、エラーの内容、考えられる原因、回復手段などがわかります。


**1** iMonitor で、アシスタントフレームの [エラー索引] をクリックします。

[エラー索引] ページからは、エラーや技術情報に関して Novell が提供している最新のドキュメントやホワイトペーパーにリンクできます。

## DSRepair 情報の表示

[DSRepair] ページでは、検出された問題を表示したり、DIB セットのバックアップやクリーンアップを実行できます。Novell iMonitor の DSRepair 機能はサーバ限定の機能です。つまり、この機能は iMonitor が動作しているサーバ以外からは起動できません。他のサーバで実行されている DSRepair 情報にアクセスするには、そのサーバで実行されている iMonitor に切り替える必要があります。eDirectory の新しいバージョンにアップグレードしたサーバの数が増えれば、iMonitor のサーバ限定機能の利用範囲は広がります。この機能以外には、[DS トレース] ページや [バックグラウンド処理スケジュール] ページなどがサーバ限定の機能です。

[DSRepair] ページの情報にアクセスするには、サーバの管理者と同等の権利またはコンソールオペレータの権利が必要です。このページの情報にアクセスするには、認証情報の確認のためにログインする必要があります。

1 iMonitor で、[DSRepair]  をクリックします。

2 次のオプションから選択します。

[ダウンロード] ファイルサーバから修復関連のファイルを取得できます。DSRepair ユーティリティが実行されている場合や、iMonitor の [DSRepair] ページから修復を開始した場合には、操作が完了するまでは dsrepair.log にアクセスできません。

[古いDIBセットを削除] 赤いXをクリックすると、古いDIBセットが削除できます。

**警告:** この操作は元に戻すことができません。このオプションを選択すると、古いDIBセットがファイルシステムからパージされます。

[DS Repair 拡張スイッチ] 問題の修正、問題のチェック、データベースのバックアップ作成などを実行できます。Novell サポートから指示がない限り、[サポートオプション] フィールドには、何も入力する必要はありません。

3 [修復の開始] をクリックして、サーバ上で DS Repair を実行します。

## エージェントのヘルス情報の表示

[エージェントヘルス] ページでは、指定した eDirectory エージェントのヘルス情報およびそのエージェントに関連するパーティションおよびレプリカリングを表示できます。

1 iMonitor で、アシスタントフレームの [エージェントヘルス] をクリックします。

2 リンクをクリックすると、詳細な情報が表示されます。

## ツリー内のオブジェクトの参照

[参照] ページでは、ユーザのツリー内にある任意のオブジェクトを参照できます。ページ最上部のナビゲーションバーに、表示中のオブジェクトが存在するサーバ、およびオブジェクトへのパスが表示されます。ページの左側にあるレプリカフレームで、実パーティション上の同じオブジェクトを表示し、これにアクセスできます。ページ内の下線付きオブジェクトをクリックすると、オブジェクトに関する詳しい情報が表示されます。また、ナビゲータフレーム内にある名前の任意の部分をクリックすると、ツリーの上の階層を参照できます。

このページに表示される情報は、ログイン時の eDirectory 権、参照するオブジェクトのタイプ、および実行している NDS または eDirectory のバージョンによって異なります。スーパーバイザ権でログインした場合、このページには XRef オブジェクトが表示されます。レプリカリストを使用して、レプリカの実コピーへジャンプできます。ダイナミックグループのオブジェクトを参照している場合、ダイナミックメンバーに対してタイムスタンプは表示されません。

[レプリカ同期] このオブジェクトを含むレプリカの同期ステータスを表示します。

[エントリ同期] サーバ側から見て同期が必要である属性を表示します。

[接続情報] iMonitor がこのオブジェクトの情報をどこで取得したかが表示されます。

[エントリ情報] オブジェクトの名前、フラグ、ベースクラス、変更タイムスタンプ、および接続情報のサマリを表示します。

[すべてのレプリカにエントリを送信] このエントリの属性を他のすべてのレプリカに再送信します。オブジェクトに多数の属性値がある場合、この処理には時間がかかることがあります。この処理では、そのオブジェクトの、他のすべてのコピーが同一になるわけではありません。他のレプリカが各属性を再検討できるようにするだけです。

[すべて送信] (参照しているオブジェクトがパーティションルートの場合および [カスタムモード] オプションが有効になっている場合のみ表示されます)。このパーティション内のすべてのエントリを、パーティションのレプリカを保持しているすべてのサーバに再送信します。この処理では、そのオブジェクトの、送信されたすべてのコピーが同一になるわけではありません。他のレプリカが各オブジェクトとその属性を再検討できるようにするだけです。

## 同期またはページのためのエントリの表示

[変更キャッシュ] ページでは、同期またはページにおいてこのサーバが検討する必要のあるエンティティのリストを表示できます。このオプションを使用できるのは、ユーザがアクセスしているサーバが eDirectory 8.6 以降を実行しており、また、表示中のオブジェクトがパーティションルートである場合だけです。このページを表示するには、NCP™ サーバに対するスーパーバイザ権が必要です。

[エントリ同期] エントリが同期を必要とする理由を判別できます。

## DirXML の詳細の表示

[DirXML の概要] ページでは、ユーザのサーバで実行中のすべての DirXML ドライバ、各ドライバのステータス、保留中の関連付け、およびドライバの詳細のリストを表示できます。

1 iMonitor で、[DirXML の概要]  をクリックします。

2 次のオプションから選択します。

[ステータス] は、指定したドライバの現在の状態を表示します。表示されるステータスは、[停止]、[開始します]、[稼働中]、[シャットダウン保留中]、および [スキーマ取得中] です。

[起動オプション] は、選択したドライバの現在の起動オプションを表示します。

[保留中] は、まだ作成されていない関連付けの数を表示します。

[ドライバ詳細] アイコンは、ユーザのサーバで実行中の DirXML ドライバに関する、加入者および発行者の詳細、XML ルール、フィルタ、および保留中の関連付けリストを表示します。このページには、最初の 50 個の保留中オブジェクトに関する詳細も表示されます。このページに表示される XML ルールを使用すると、指定した DirXML ドライバに対するオブジェクトの作成を続行するために必要な、保留中のオブジェクト内で検索すべき情報を判断できます。



## レプリカの同期ステータスの表示

[レプリカ同期] ページでは、レプリカの同期ステータスを表示できます。

- 1 iMonitor で、アシスタントフレームの [エージェント同期] をクリックします。
- 2 表示するパーティションの [レプリカ同期] をクリックします。
- 3 このページにあるリンク、および左側のナビゲーションバーにあるリンクを使用すると、他のパーティションにアクセスしたり、レプリカリング内でジャンプすることができます。

## レポートの設定と表示




[レポート] ページでは、このサーバで直接実行されているレポートを表示および削除できます。一部のレポートでは、実行に長時間を要し、多くのリソースを消費する場合があります。

スケジュール設定されたレポートは、ユーザ認証なし、つまり [Public] で実行します。ユーザが実行するレポートはすべて、ユーザの権利で直接実行されます。すべてのレポートデータは、そのレポートの実行元サーバに格納されます。



[レポートの環境設定] ページでは、事前に設定されたレポート、カスタムレポート、およびスケジュール設定されたレポートのリストを表示できます。このページを使用して、レポートを変更および実行できます。また、iMonitor ページ用のカスタムレポートの作成もできます。次の表では、iMonitor 2.1 であらかじめ設定されているレポートを表示します。

レポート	説明
サーバ情報	ツリー全体を調べて、検索可能な各 NCP サーバと通信し、検知したすべてのエラーをレポートします。このレポートを使用して、時刻同期およびリンバの問題を診断できます。また、現在のサーバ自体が他のすべてのサーバと通信可能であるかどうかも知ることができます。環境設定ページで選択されている場合、このサーバはツリー内にある各サーバのNDSエージェントヘルス情報を生成することもできます。
破損通知リスティング	このサーバ上のすべての破損通知を表示します。
オブジェクト統計	オブジェクトを指定したスコープで調べて、要求される条件に一致したオブジェクトのリストを生成します。この条件には、将来の時刻、不明なオブジェクト、名前が変更されたオブジェクト、ベースクラス数、コンテナ、別名、外部参照などがあります。
サービスアドバタイズ	SLP または SAP を使用して現在のサーバに認識されている、すべてのディレクトリとサーバを表示します。
エージェントヘルス	現在のサーバのヘルス情報を収集します。



### レポートの表示や削除を行う

- 1 iMonitor で、[レポート]  をクリックします。
- 2 レポートを削除する場合は  を、レポートを表示する場合は  をクリックします。

### レポートを実行する



- 1 iMonitor で、[レポート]  > [レポート設定] の順にクリックします。
- 2  をクリックして、レポートを実行します。

## レポートの設定またはスケジューリングを行う

- 1 iMonitor で、[レポート]  > [レポート設定] の順にクリックします。
- 2  をクリックして、レポートを設定またはスケジュールします。
- 3 目的のオプションを選択し、[デフォルトの保存] をクリックして選択したオプションを保存します。
- 4 (オプション)レポートを定期的に行うか、または後で実行するかを設定します。
  - 4a レポートの頻度、開始時刻、および開始日を指定します。
  - 4b [スケジュール] をクリックします。
- 5 [レポートの実行] をクリックして、レポートを開始します。

## カスタムレポートを作成する

カスタムレポートを作成すると、iMonitor の任意のページをレポートとして起動できます。

- 1 iMonitor で、[レポート]  > [レポート設定] の順にクリックします。
- 2 [実行可能レポートリスト] の [カスタムレポート] 行にある  をクリックします。
- 3 レポートの名前を入力し、レポートとして起動する iMonitor ページの URL を入力します。

カスタムレポートを実行する場合は、次の URL を入力します。

***/nds/required page***
- 4 保存するレポートのバージョンの数を指定します。
- 5 (オプション) [保存] をクリックして、レポートを保存します。
- 6 (オプション)レポートを定期的に行うか、または後で実行するかを設定します。
  - 6a レポートの頻度、開始時刻、および開始日を指定します。
  - 6b [スケジュール] をクリックします。
- 7 [レポートの実行] をクリックして、レポートを開始します。

## スキーマ、クラス、および属性定義の表示

[スキーマ] ページでは、ユーザのスキーマ、クラス、および属性の定義を表示できます。すでに作成されている拡張や特定のスキーマに固有の情報 (スキーマに行った変更や拡張など) を添付して、ツリー上にロードされているスキーマを表示できます。

- 1 iMonitor で、アシスタントフレームの [スキーマ] をクリックします。
- 2 次のオプションから選択します。

[同期リスト] このサーバと同期する相手のサーバを表示します。このオプションは、NDS eDirectory8.5 以降を実行しているサーバに対してのみ使用できます。この情報を表示するには、サーバに対するスーパーバイザ権が必要です。

[スキーマルート] ツリーのルートに最も近いスキーマレプリカに関する情報を表示します。

各 eDirectory サーバには、エントリ内のスキーマのレプリカが保存されています。スキーマレプリカは、ディレクトリオブジェクトを格納しているパーティションから分割されて保存されます。任意のスキーマレプリカへの変更内容は、すべてのレプリカに伝えられます。スキーマの変更は、ルートパーティションの書き込み可能な

レプリカを保存するサーバを通してのみ実行できます。ルートパーティションの読み込み可能なレプリカを保存しているサーバは、スキーマ情報を読み込むことはできませんが、変更はできません。


**[属性定義]** 各属性の名前、属性値が含まれる構文、および属性が受ける制約を表示します。左側のナビゲーションフレームを使用すると、個々の属性を参照したり、それらにアクセスすることができます。

**[クラス定義]** 各クラスの名前、ルール、および属性を表示します。左側のナビゲーションフレームを使用すると、個々の属性を参照したり、それらにアクセスすることができます。

## オブジェクトの検索

**[検索]** ページでは、さまざまなクエリオプションおよびフィルタに基づいて、オブジェクトを検索できます。検索クエリオプションおよびフィルタは、基本フォームとカスタムフォームという2つのレベルの検索リクエストフォームに分けられます。基本検索リクエストフォームは、eDirectoryの一般ユーザ向けであり、基本的な検索に使用します。カスタム検索リクエストフォームは、熟練ユーザ向けであり、複雑な検索に使用します。現在はサーバレベルの検索のみがサポートされています。

4つのセクション内の検索オプションおよび検索フィルタは、すべて結合可能です。空白フィールド(相対識別名を除く)は無視されます。<Ctrl>キーを使用して、マルチリスト上でアイテムを選択解除したり、複数のアイテムを選択することができます。選択解除したマルチリストも無視されます。

**1** Novell iMonitor で、[検索]をクリックします。

**2** 次のオプションから選択します。

**[スコープオプション]** 検索のスコープを指定できます。

**[エントリフィルタ]** エントリ情報に関連する検索クエリフィルタを指定できます。

**[属性と値のフィルタ]** 属性および値に関連する検索クエリフィルタを指定できます。

**[表示オプション]** 検索結果の表示形式を制御するオプションを指定できます。

**3** 検索リクエストフォームの一番下にある [ヘルプ] ボタンをクリックすると、そのフォームに関連する簡単なヘルプ情報が表示されます。

ヘルプ情報を消すには、[再ロード] または [更新] をクリックします。

## ストリームビューアの使用

**[ストリームビューア]** ページでは、次の形式で現在のストリームを表示できます。

- ◆ プレーンテキスト
- ◆ HTML
- ◆ GIF
- ◆ JPEG
- ◆ BMP
- ◆ WAV
- ◆ 16進ダンプ
- ◆ その他

特定の形式で常に表示したいストリーム属性がある場合、[ストリームビューア] を使用してデフォルトの表示設定を選択します。

[NDS ストリーム属性セットアップ] ブラウザでストリームを表示するためのデフォルトの形式を変更します。ストリームが正しく表示されるかどうかはブラウザに依存します。ブラウザによっては、ユーザが選択した設定が適用されない場合があります。

デフォルト設定に加えた変更を適用するには、ユーザがサーバに認証される必要があります。変更内容は streams.ini(NetWare および Windows サーバの場合) または streams.conf(Solaris および Linux サーバの場合) に格納されるため、デフォルトの設定は手動で編集することもできます。

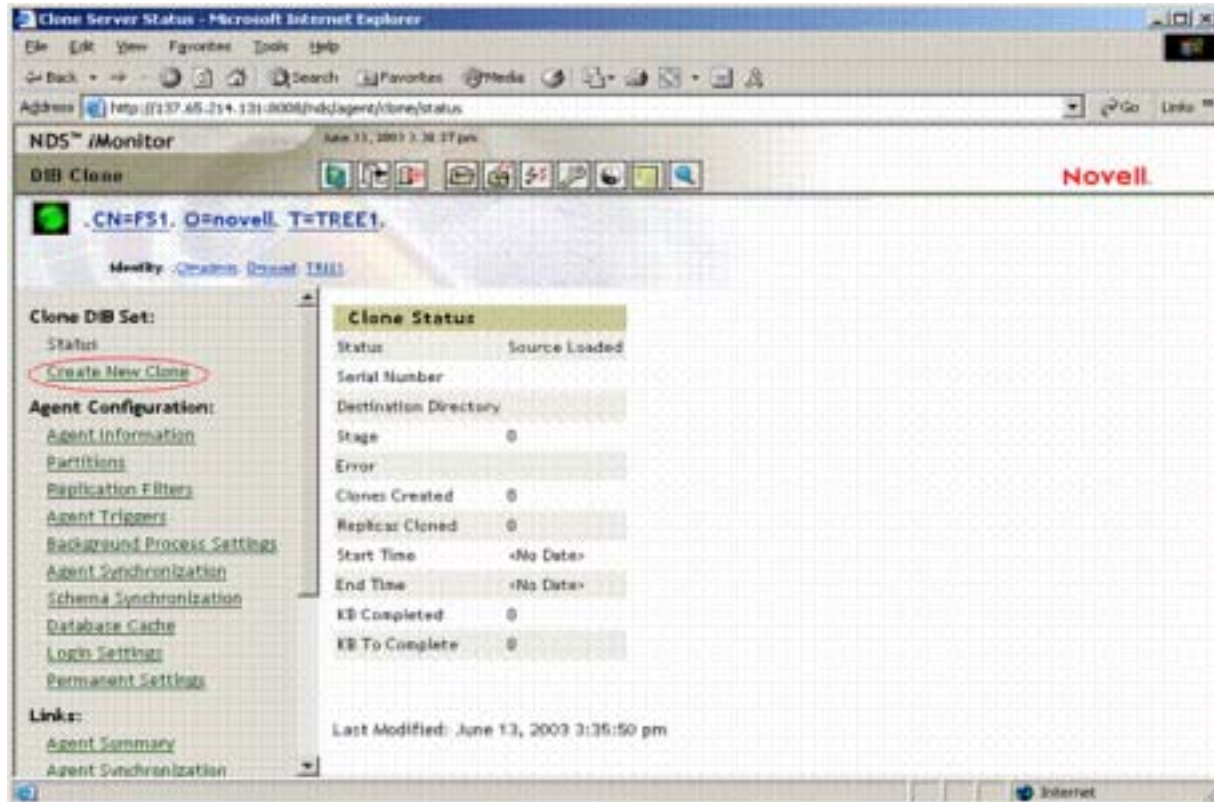
## DIB セットのクローン

このオプションでは、ツリーおよびすべてのオブジェクトの完全な DIB ファイルセットを複製します。クローンは別のサーバに配置することができます。インポート先サーバが eDirectory を開始すると、サーバは DIB ファイルセットをロードし、ツリーのマスタ (オリジナル) レプリカに接続し、名前を解決し、クローン作成後に行われた DIB ファイルセットのすべての変更を同期します。

eDirectory DIB セットのクローンは、クローンを作成したサーバのオペレーティングシステムと同じオペレーティングシステムが稼働するサーバ上にだけ配置する必要があります。たとえば、DIB ファイルセットのクローンを Solaris サーバに復元する場合、NetWare や Windows サーバではなく、Solaris サーバ上でそのクローンを作成します。

この機能のバックエンドは eDirectory 8.7 に搭載されていましたが、eDirectory 8.7.1 で iMonitor 2.1 以降が稼働するようになるまでサポートされていませんでした。このオプションは、バージョンが 8.7 以前の Novell eDirectory や NDS では使用できません。

図 28 iMonitor の [DIB セットのクローン] ページ



このセクションでは、次の情報について説明します。

- ◆ [193 ページの「DIB セットのクローンの使用事例」](#)
- ◆ [193 ページの「クローンを作成する」](#)

## DIB セットのクローンの使用事例

DIB セットのクローンは次のような場合に使用します。

- ◆ すでに「オン」の状態になっているパーティションで新しいサーバを作成します。  
次の利点があります。
  - ◆ レプリカリングに新しいサーバを追加する際に、リング内のすべてのサーバが稼動中または実行中である必要がありません。
  - ◆ 新しいサーバは自動的にすべてのパーティションを保持しますが、同期する必要はありません。
  - ◆ すばやく処理できます。
- ◆ 障害回復

利点	欠点
パーティションを 1 度コピーするだけで成功します。	対象のパーティションを少なくとも1度は正しくコピーする必要があります。
複数のパーティションをもつ大きなサーバを停止させる時間が少なくて済みます。	SSL またはセキュリティバックアップを利用しません。  ファイルシステムを処理しません。

- ◆ バックアップおよび復元

利点	欠点
大規模のデータベースでは特に、処理に時間がかかりません。	eDirectory のコアを追加するだけです。LDAP、SNMP、SSL などはインストールされません。また設定も行われません。  最新の変更は取得されません。スナップショットのみが取得されます。ロールフォワードログは実行されません。

このような欠点があるため、バックアップ処理および復元処理のために DIB セットのクローンを使用することはお勧めできません。

## クローンを作成する

DIB ファイルセットのクローンは、元のサーバでオンラインまたはオフラインのいずれでも作成できます。オフラインで行う場合は、eDirectory を停止させておく必要があります。オンラインでは、eDirectory はロックされません。

- ◆ [194 ページの「オンラインによる方法」](#)
- ◆ [195 ページの「オフラインによる方法」](#)

## オンラインによる方法

### 1 ツリーのスキーマを拡張します。

スキーマは必ず拡張してください。拡張しない場合、エラーが発生します。eDirectory のインストール時に含まれている `dibclone.sch` を使用します。これにより、iMonitor クローンユーティリティの操作に必要な属性が追加されます。

プラットフォーム	スキーマを拡張するには、次の操作を行います。
NetWare	NWConfig を使用します (NWConfig.nlm> [環境設定] オプション> [ディレクトリ] オプション> [スキーマの拡張] の順に実行します)。  dibclone.sch は <code>sys:\system\schema</code> にあります。
Windows	NDSCons.exe を使用します (NDSCons.exe により <code>install.dlm</code> をロードし、次に [追加のスキーマファイルのインストール] をクリックします)。  dibclone.sch は <code>C:\Novell\NDS</code> にあります。
Linux、Solaris、AIX、HP-UX	ndssch を使用します。  dibclone.sch は <code>/usr/lib/nds-schema</code> にあります。  詳細については、111 ページの「 <a href="#">ndssch ユーティリティを使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張する</a> 」を参照してください。

### 2 DIB ファイルセットのクローンを作成します。

#### 2a クローンを作成するサーバに dsclone モジュールをロードします。

プラットフォーム	モジュール
NetWare	サーバコンソールで、「 <code>dsclone.nlm</code> 」と入力します。
Windows	NDSCons.exe で <code>dsclone.dll</code> を選択し、[開始] をクリックします。
Linux、Solaris、AIX、HP-UX	「 <code>ndsclone</code> 」 エントリを <code>ndsmodules.conf</code> ファイルに追加し、次に <code>http://IP address:port/dhost</code> ページを使用して、Directory Clone Agent をロードします。

#### 2b iMonitor で、DIB 環境設定のクローンを実行します。

[エージェント環境設定] > 「DIB セットのクローン」 > [新しいクローンの作成] の順にクリックします。

#### 2c クローンが作成されるサーバの完全修飾名と、DIBファイルのクローンが配置される場所のファイルパスを指定し、次に [クローンオブジェクトを作成] チェックボックスと [DIBをオンラインでクローン] チェックボックスをオンにします。

#### 2d [送信] をクリックします。

NDS クローンオブジェクトが作成され、DIB ファイルセットが指定された配置先にコピーされます。

- 3 DIB ファイルセットのクローンを適切なディレクトリにある受信側のサーバ上に移動します。さらに、`/etc/nds.conf file` を受信側のサーバに転送し、ファイル内のすべての参照を受信側のサーバ名で更新します。
- 4 クローンサーバ上で eDirectory を起動します。

クローンサーバオブジェクトのマスタレプリカが eDirectory を実行しており、使用できることを確認します。クローンサーバ上で eDirectory が初期化されると、eDirectory はクローンサーバの最終的な名前が解決できるマスタレプリカと通信を行います。

## オフラインによる方法

- 1 ツリーのスキーマを拡張します。

スキーマは必ず拡張してください。拡張しない場合、エラーが発生します。eDirectory のインストール時に含まれている `dibclone.sch` を使用します。これにより、iMonitor クローンユーティリティの操作に必要な属性が追加されます。

プラットフォーム	スキーマを拡張するには、次の操作を行います。
NetWare	NWConfig を使用します (NWConfig.nlm> [環境設定] オプション> [ディレクトリ] オプション> [スキーマの拡張] の順に実行します)。  dibclone.sch は <code>sys:\system\schema</code> にあります。
Windows	NDSCons.exe を使用します (NDSCons.exe により <code>install.dlm</code> をロードし、次に [追加のスキーマファイルのインストール] をクリックします)。  dibclone.sch は <code>C:\Novell\NDS</code> にあります。
Linux、Solaris、AIX、HP-UX	ndssch を使用します。  dibclone.sch は <code>/usr/lib/nds-schema</code> にあります。  詳細については、 <a href="#">111 ページの「ndssch ユーティリティを使用して、Linux、Solaris、AIX、または HP-UX システム上のスキーマを拡張する」</a> を参照してください。

- 2 DIB ファイルセットのクローンを作成します。
  - 2a iMonitor で、DIB 環境設定のクローンを実行します。  
[エージェント環境設定] > 「DIB セットのクローン」 > [新しいクローンの作成] の順にクリックします。
  - 2b クローンが作成されるサーバの完全修飾名を指定し、[クローンオブジェクトを作成] チェックボックスをオンにします。
  - 2c [送信] をクリックします。  
NDS クローンオブジェクトが作成されます。eDirectory は停止しているため、eDirectory がロックされているというエラーが報告されます。
  - 2d 関連するすべてのディレクトリとファイルも含めた完全な DIB ファイルセットを、配置先または配置先サーバに DIB ファイルセットを移動するために便利なメディアに手動でコピーします。
  - 2e 元のサーバ上で、eDirectory を起動します。

- 3 DIB ファイルセットのクローンを適切なディレクトリにある受信側のサーバ上に移動します。
- 4 クローンサーバ上で eDirectory を起動します。

クローンサーバオブジェクトのマスタレプリカが eDirectory を実行しており、使用できることを確認します。クローンサーバ上で eDirectory が初期化されると、eDirectory はクローンサーバの最終的な名前が解決できるマスタレプリカと通信を行います。

## セキュリティ保護された iMonitor 操作の実現

iMonitor 環境へのアクセスをセキュリティ保護するには、次の保護手順を実行します。

1. ファイアウォールを使用して VPN アクセスを準備します。これは、Novell iManager および、アクセス制限が必要な他のすべての Web ベースのサービスの場合も同様です。
2. ファイアウォールが設置されているかどうかに関係なく、アクセスの種類を制限することによって、iMonitor はさらに DoS(Denial of Service) 攻撃から保護されます。

iMonitor は URL 要求を経由して受け取るデータを十分に確認しますが、あらゆる不正な入力を拒否できるとは保証できません。無効な URL を通じた DoS 攻撃の危険を減らすため、**iMonitor** の環境設定ファイルを通じて 3 つのレベルのアクセスが LockMask: オプションを使用して制御されます。

アクセスレベル	説明
0	iMonitor の URL 処理において事前の認証は不要です。この場合、[Public] 識別子の eDirectory 権利がすべての要求に適用され、iMonitor が表示する情報は [Public] ユーザの権利に限定されます。ただし、iMonitor に URL を送る際に認証が不要であるため、iMonitor は、不正な URL の送信による DoS 攻撃を受けやすくなる可能性があります。
1(デフォルト)	iMonitor が URL を処理する前に、eDirectory 識別子としての認証が必要です。この場合、その識別子の eDirectory 権利はすべての要求に適用されるため、eDirectory 権利によって制限を受けます。DoS 攻撃を受ける危険性はレベル 0 と同様ですが、DoS 攻撃は実際にサーバに認証を受けたものでない行うことができないことが異なります。認証が正常に実行されるまでは、すべての iMonitor の URL 要求への返答は [ログイン] ダイアログボックスで行われます。したがってこの段階では、設定された正当性を持たないユーザによる攻撃を通さないようにする必要があります。
2	iMonitor が URL を処理する前に、iMonitor が認証しているサーバ上のスーパーバイザに相当する eDirectory 識別子としての認証が必要です。DoS 攻撃を受ける危険性はレベル 1 と同様ですが、DoS 攻撃を行うには実際にサーバのスーパーバイザとして認証される必要があります。認証が正常に実行されるまでは、すべての iMonitor の URL 要求への返答は [ログイン] ダイアログボックスで行われます。したがってこの状態に設定されているときには、iMonitor は認証されていないユーザおよびスーパーバイザとして認証されていないユーザからの攻撃を通さないようにする必要があります。

レベル 1 はデフォルトです。多くの管理者はツリー内のすべてのサーバにアクセスできるスーパーバイザ権を持っていませんが、管理しているサーバと通信するサーバ上の iMonitor サービスを使用する必要が生じる可能性があるためです。

注：iMonitor には Repair、トレースなど複数の機能があり、これらの機能にアクセスするには、LockMask の設定に関係なくスーパーバイザに相当する権利が必要です。



# 8

## Novell eDirectory ツリーのマージ

Novell® eDirectory™ マージユーティリティを使用すると、2つの Novell eDirectory ツリーをマージして、単一の eDirectory ツリーを作成できます。マージされるのはツリーオブジェクトだけです。コンテナオブジェクトとそのリーフオブジェクトは、マージ後のツリー上でもそれぞれ異なるオブジェクトとして存在します。

**ヒント:** リーフオブジェクトを移動する場合や、パーティションをマージする場合には、ConsoleOne® または Novell iManager を使用します。

マージする 2 つのツリーをそれぞれソースツリーおよびターゲットツリーとといいます。1 つのツリーを別のツリーにマージする前に、ターゲットツリーにあるルートパーティションのレプリカを 1 つだけ残し、その他のすべてのレプリカを削除する必要があります。ターゲットツリーにルートパーティションのレプリカが 1 つしかない場合には、マージ処理を続行できます。マージ後は、ルートパーティションのレプリカが 2 つになります。1 つはターゲットツリー上にあったレプリカで、もう 1 つはマージ操作を実行したソースツリーのサーバ上にあったレプリカです。ルートパーティションのレプリカを追加する必要がある場合は、マージが完了した後に保存することができます。

マージ時にターゲットツリーサーバにルートパーティションのレプリカが複数ある場合、マスタレプリカを保持していないサーバで、外部参照オブジェクトの位置に関する問題が発生する可能性があります。外部参照オブジェクトは、サブオーディネートリファレンスのパーティションルートに含まれています。これは、パーティションの境界を表すルートパーティションのレプリカを持つ別のサーバに配置する必要があります。ソースツリーにあるルートパーティションの下位パーティションごとに、ターゲットツリーにあるサブオーディネートリファレンスのパーティションルートを持つ必要があります。エラーが生じた場合、同期ステータスに関する eDirectory エラーコード -605 が報告されます。この場合、DSRepair を使用して、エラーが発生したサーバのローカルデータベースを修復します。詳細については、[217 ページの「ローカルデータベースの修復の実行」](#)を参照してください。

DSMerge を実行しても、コンテナ内の eDirectory 名またはコンテキストは変わりません。オブジェクト権とプロパティ権はマージ後のツリーでも保持されます。

この章では、次のトピックについて説明します。

- ◆ [197 ページの「eDirectory ツリーのマージ」](#)
- ◆ [203 ページの「サーバツリーの結合」](#)
- ◆ [209 ページの「ツリー名の変更」](#)

## eDirectory ツリーのマージ

eDirectory ツリーをマージするには、Novell iManager の [ツリーのマージ] ウィザードを使用します。このウィザードでは、2 つの eDirector ツリーのルートをマージできます。マージされるのはツリーオブジェクトだけです。コンテナオブジェクトとそのリーフオブジェクトは、マージ後のツリー上でもそれぞれ異なるオブジェクトとして存在します。

マージする 2 つのツリーはそれぞれソースツリーおよびターゲットツリーとといいます。ソースツリーのマージ先ツリーがターゲットツリーです。

DSMerge を実行しても、コンテナ内のオブジェクトの名前は変わりません。オブジェクト権とプロパティ権はマージ後のツリーでも保持されます。

- ◆ 198 ページの「前提条件」
- ◆ 198 ページの「ターゲットツリーの要件」
- ◆ 199 ページの「ソースツリーをターゲットツリーへマージする」
- ◆ 199 ページの「パーティションの変化」
- ◆ 200 ページの「ソースツリーとターゲットツリーを準備する」
- ◆ 201 ページの「マージする前の時刻の同期」
- ◆ 201 ページの「2 つのツリーをマージする」
- ◆ 203 ページの「マージ後の作業」

## 前提条件


- ❑ Novell eDirectory 8.7.3がソースツリーの[Root]パーティションのマスタレプリカを含むサーバにインストールされている必要があります。
- ❑ 正しい機能を維持するには、ソースツリーの他のサーバを eDirectory 8.6 以降にアップグレードする必要があります。

## ターゲットツリーの要件

- ❑ Novell eDirectory 8.7.3がターゲットツリーの[Root]パーティションのマスタレプリカを含むサーバにインストールされている必要があります。このサーバで他のバージョンの NDS<sup>®</sup> または eDirectory が実行されている場合は、マージ操作が正常に完了されません。
- ❑ 正しい機能を維持するには、ターゲットツリーの他のサーバを eDirectory 8.6 以降にアップグレードする必要があります。
- ❑ ソースツリーとターゲットツリーの両方で、ツリーの従属コンテナを同じ名前で維持することはできません。2 つのツリーをマージする前に、一方のコンテナをリネームする必要があります。
- ❑ ソースツリーとターゲットツリーの両方にセキュリティオブジェクトがある場合は、ツリーをマージする前にどちらかを削除する必要があります。

## スキーマの要件

マージ操作を実行する前に、2 つのツリーのスキーマが正確に一致している必要があります。ツリーごとに [Root] パーティションのマスタレプリカを含むサーバで DSRepair を実行する必要があります。[リモートスキーマのインポート] オプションを使用して、各ツリーで他のツリーのスキーマがすべて認識されていることを確認します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [スキーマの保守] の順にクリックします。
- 3 スキーマメンテナンス操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 指定したサーバへの認証を行い、[次へ] をクリックします。
- 5 [リモートスキーマのインポート] > [次へ] の順にクリックします。
- 6 スキーマのインポート元のツリー名を指定します。

7 [開始] をクリックします。

このオプションは、ソースツリーとターゲットツリーの両方でスキーマの違いが報告されなくなるまで実行する必要があります。そうでないとマージ操作は成功しません。

8 スキーマメンテナンス操作から戻された情報とともに「完了」メッセージが表示されたら、[閉じる] をクリックして終了します。

## ソースツリーをターゲットツリーへマージする

ツリーをマージすると、ソースツリー内のサーバがターゲットツリーに組み込まれます。

ターゲットツリーのツリーオブジェクトがソースツリー内のオブジェクトの新しいツリーオブジェクトになり、ソースツリーにあるすべてのサーバのツリー名がターゲットツリーのツリー名に変わります。

ターゲットツリー内のサーバのツリー名はマージ後も変わりません。

ソースツリーのツリーオブジェクトの下位オブジェクトは、ターゲットツリーのツリーオブジェクトの下位オブジェクトになります。

## パーティションの変化

マージ実行時には、ソースツリーのツリーオブジェクトの下にあるオブジェクトがDSMergeによって複数のパーティションに分けられます。

続いて、ソースツリー内のサーバから、ツリーパーティションのレプリカがマスタレプリカを除いてすべて削除されます。ソースツリーのマスタレプリカを保持していたサーバには、ターゲットツリーのツリーパーティションのレプリカが渡されます。

図 29 および図 30 は、2つのツリーをマージした場合のパーティションの変化を示します。

図 29 マージ前の eDirectory ツリー

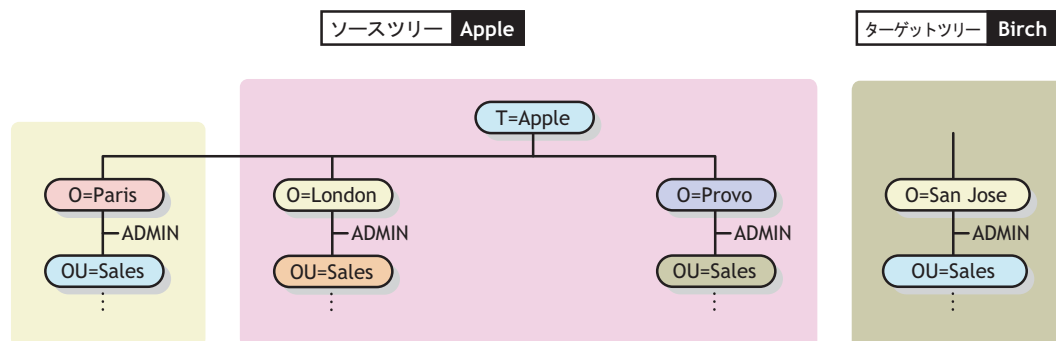
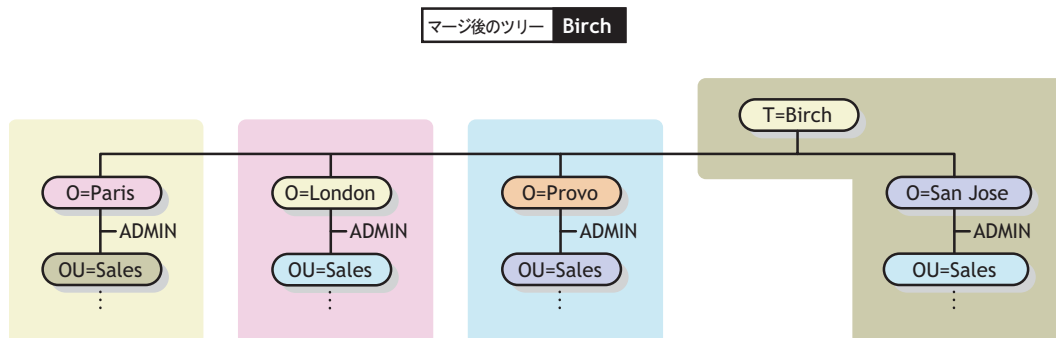


図 30 マージ後の eDirectory ツリー



## ソースツリーとターゲットツリーを準備する

マージ操作を開始する前に、そのマージ操作の影響を受けるすべてのサーバが安定した状態で同期していることを確認する必要があります。次の表で、マージ対象のソースツリーとターゲットツリーの準備に関する前提条件について説明します。

前提条件	必要な措置
ソースツリーまたはターゲットツリーのツリーパーティションのレプリカを保持しているすべてのサーバ上で WANMAN がオフになっている。	WANMAN ポリシーを調べて、WAN 通信の制約によるマージ操作への支障がないことを確認します。必要があれば、マージ操作を開始する前に WANMAN をオフにします。
ソースツリーのツリーオブジェクトに別名またはリーフオブジェクトが存在しない。	ソースツリーのツリーオブジェクトに存在する別名またはリーフオブジェクトを削除します。
ソースツリーおよびターゲットツリーに同じ名前がない。	同じ名前がある場合には、ソースツリーまたはターゲットツリーのオブジェクト名を変更します。コンテナオブジェクト名を変更したくない場合は、コンテナの 1 つから同じツリー内の別のコンテナにオブジェクトを移動し、空になったコンテナを DSMerge 実行前に削除します。詳細については、 <a href="#">3 章 89 ページの「オブジェクトの管理」</a> を参照してください。  ツリーオブジェクトの直下でないコンテナオブジェクトの場合は、両ツリーに同じコンテナオブジェクトがあってもかまいません。
ソースツリー上にログイン接続が存在しない。	ソースツリーのすべての接続を終了します。
ソースツリーの eDirectory およびターゲットツリーの eDirectory のバージョンが同じである。	ルートパーティションのレプリカを持つ、eDirectory 8.7.3 でないすべてのサーバをアップグレードします。
ターゲットツリー内のルートレプリカのコピーが 1 つである。	ターゲットツリー上の、マスターレプリカを除いたすべてのレプリカを削除します。
ソースツリーおよびターゲットツリーのスキーマが同じである。	DSMerge を実行します。出力されるレポートを見てスキーマに問題があることがわかった場合は、DSRepair を使用してスキーマを一致させます ( 詳細については、 <a href="#">228 ページの「リモートスキーマをインポートする」</a> を参照してください)。DSMerge を再実行します。
単一のツリーだけが、ツリーのルートの下位にセキュリティコンテナを保持できる。	ソースツリーおよびターゲットツリーの両方がセキュリティコンテナを保持している場合は、 <a href="#">553 ページの付録 A、「NMAS の注意事項」</a> で説明されている手順に従って、一方のコンテナを削除します。

マージ操作は単独のトランザクションなので、実行中に停電やハードウェアエラーが発生しても重大な障害にはつながりません。ただし、DSMerge を実行する前に、あらかじめ eDirectory データベースの通常のバックアップをとっておくことをお勧めします。詳細については、[14 章 381 ページの「Novell eDirectory のバックアップと復元」](#)を参照してください。

## マージする前の時刻の同期

**重要：**時刻同期の正確な環境設定は複雑な作業です。ツリーをマージする前に、両ツリーを同期させるのに十分な時間があることを確認します。

時刻の異なる複数のタイムソースが使用されていたり、ツリー内のサーバの中に時刻が同期されていないものがあると、Novell eDirectory は正しく機能しません。

マージを実行する前に、両ツリーにあるすべてのサーバの時刻が同期されていること、およびこれらのサーバがタイムソースとして同じタイムサーバを使用していることを確認します。ただし、ターゲットツリーの時刻は 5 分以内であればソースツリーの時刻より進んでいてもかまいません。

一般に、1 つのツリー上に存在できるタイムサーバは、リファレンスタイムサーバまたはシングルタイムサーバのどちらか 1 つだけです。同様に、マージ後のツリー上に存在できるタイムサーバも、リファレンスタイムサーバまたはシングルタイムサーバのどちらか 1 つだけです。

マージする両ツリーのそれぞれにリファレンスタイムサーバまたはシングルタイムサーバがある場合は、どちらかのツリーの設定をもう一方のツリーにあるリファレンスタイムサーバまたはシングルタイムサーバに変更して、マージ後のツリー上でリファレンスタイムサーバまたはシングルタイムサーバが 1 つだけになるようにします。

タイムサーバの種類の詳細については、『[Network Time Management Administration Guide \(Network Time Management 管理ガイド\)](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html) ([http://www.novell.com/documentation/japanese/nw65/time\\_enu/data/hl5k6r0y.html](http://www.novell.com/documentation/japanese/nw65/time_enu/data/hl5k6r0y.html))』を参照してください。

## 2 つのツリーをマージする

すべてのメニューオプションの機能を使用できるようにするには、ツリーパーティションのマスタレプリカが格納されているサーバ上で DSMerge を実行します。

マスタレプリカが格納されている場所が不明な場合は、マスタレプリカを必要とする操作を実行すると、正確なサーバ名とともに表示されます。

マージ操作を実行するには、次のどちらかの方法を使用します。

- ◆ Novell iManager
- ◆ eMBox コマンドラインクライアント

詳細については、[210 ページの「eMBox クライアントを使用したツリーのマージ」](#)を参照してください。

大きなツリーをマージするときは、ツリーオブジェクト直下にあるオブジェクトの数が少ないほうのツリーをソースツリーとして指定したほうが、処理速度が大幅に速くなります。ツリーオブジェクトの直下にあるオブジェクトすべてに対して新しいパーティションが作成されるため、この方法をとると、マージ実行時に分割されるパーティションの数が少なくてすむからです。

ソースツリーの名前はマージ後には存在なくなります。したがって、場合によっては、クライアントワークステーションの環境設定を変更する必要があります。Novell Client™ for DOS/Windows を使用している場合は、net.cfg ファイルの優先ツリーステートメントおよび優先サーバステートメントを確認します。Novell Client for Windows NT/2000 または Novell Client for Windows 95/98 を使用している場合は、クライアントのプロパティページにある優先ツリーステートメントおよび優先サーバステートメントを確認します。

優先サーバが使用されている場合は、ツリーのマージやツリー名の変更を行っても、そのクライアントは名前によってサーバにログインした状態のままなので操作による影響はクライアント側にはありません。優先ツリーが使用されている場合は、ツリーのマージやツリー名の変更を行うと、元のツリー名はなくなります。マージを行った後にはターゲットツリーの名前だけが残ります。優先ツリーの名前を新しいツリー名に変更します。

**ヒント**：ターゲットツリーの名前はマージの結果作成されるツリーの名前としてそのまま残るので、アップデートする必要のあるクライアントワークステーションの数を少なくするには、クライアントワークステーションの数が多い方のツリーをターゲットツリーとして指定します。または、マージ操作の後でツリー名を変更して、最終的なツリー名が接続されているクライアントワークステーションの数が多い方のツリーに一致するようにします。詳細については、[209 ページの「ツリー名の変更」](#)を参照してください。


次の前提条件の一覧を使用して、マージ操作の準備ができていないか確認します。

- ❑ iManager によるソースツリーのサーバへのアクセス権があること
- ❑ マージする両ツリーのツリーオブジェクトに対するスーパーバイザオブジェクト権を持つ管理者オブジェクトの名前とパスワードがわかっていること
- ❑ 2 つのツリーの eDirectory データベースがバックアップされていること
- ❑ 両ツリー内のすべてのサーバが同期されていて、同じタイムソースを使用していること
- ❑ (オプション)ツリー内のサーバがすべて動作可能であること(動作不能状態のサーバは動作可能になった時点で自動的に更新されます)。
- ❑ [200 ページの「ソースツリーとターゲットツリーを準備する」](#)に表示された前提条件を確認すること

マージプロセス自体には 2、3 分しかかかりませんが、次のような場合は、付随的な作業が必要になるため、マージ操作が完了するまでに要する時間が長くなります。

- ◆ ツリーオブジェクトの下に多数のオブジェクトがあり、パーティションに分ける必要がある
- ◆ ソースツリーに多数のサーバがあり、ソースツリーのツリー名を変更する必要がある

2 つのツリーをマージするには、次を実行します。

- 1** Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2** [eDirectory の保守] > [ツリーのマージ] の順にクリックします。
- 3** ソースツリーとしてマージを実行するサーバを指定し、[次へ] をクリックします。
- 4** サーバへの認証を行ってから、[次へ] をクリックします。
- 5** ソースツリーの管理者ユーザ名とパスワードを指定します。
- 6** ターゲットツリー名、管理者ユーザ名、およびパスワードを指定して [開始] をクリックします。  
[ツリーのマージ] ウィザードのステータスウィンドウが表示され、マージの進行状況が表示されます。
- 7** マージプロセスから戻された情報とともに「完了」メッセージが表示されたら、[閉じる] をクリックして終了します。

## マージ後の作業

2つのツリーをマージした後は、状況に応じて次の手順を行ってください。

- 1 すべてのツリー名が正しく変更されたことを確認します。
- 2 マージ操作によって作成された新しいパーティションを確認します。

新しいツリーに小さいパーティションが多数存在する場合や、関連する情報が格納されたパーティションが複数存在する場合は、これらのパーティションをマージすることもできます。詳細については、[119 ページの「パーティションのマージ」](#)を参照してください。

- 3 DSmerge を実行する前にサーバを NetWare 5 にアップグレードしなかった場合は、NetWare 5 以外を実行しているすべてのサーバに新しいレプリカをコピーします。
- 4 DSmerge 実行前にリーフオブジェクトや別名をツリーから削除した場合は、ツリーにそれらのリーフオブジェクトや別名を再作成します。
- 5 eDirectory ツリーのパーティション構成を調べます。  
マージ後のツリー内でのレプリカの位置はソースツリー内での位置と異なる場合があります。パーティション構成の変更や検査は慎重に行ってください。
- 6 クライアントワークステーションの環境設定を更新します。

Novell Client for DOS/Windows を使用している場合は、net.cfg ファイルの優先ツリーステートメントおよび優先サーバステートメントを確認します。Novell Client for Windows NT/2000 または Novell Client for Windows 95/98 を使用している場合は、クライアントのプロパティページの優先ツリーステートメントおよび優先サーバステートメントを確認するか、またはターゲットツリーの名前を変更します。

優先サーバが使用されている場合は、ツリーのマージやツリー名の変更を行っても、そのクライアントは名前によってサーバにログインした状態のままなので操作による影響はクライアント側にはありません。優先ツリーが使用されている場合は、ツリーのマージやツリー名の変更を行うと、元のツリー名はなくなります。マージを行った後にはターゲットツリーの名前だけが残ります。優先ツリーの名前を新しいツリー名に変更します。

ソースツリーのツリーオブジェクトの ACL(アクセス制御リスト) は保持されます。したがって、このツリーオブジェクトに対してソースツリーの管理者ユーザが持つ権利は、操作後も有効です。

マージ後も、両ツリーの管理者ユーザは存在したままで、それぞれのコンテナオブジェクトによって固有のものとして識別されます。

セキュリティ上必要であれば、2つの管理者ユーザオブジェクトの一方を削除するか、両オブジェクトの権利を制限することもできます。

## サーバツリーの結合

[ツリーの結合] オプションを使用すると、単一サーバソースツリーのツリーオブジェクトを、ターゲットツリーにある指定のコンテナに結合できます。結合が完了すると、ソースツリーのツリー名は、ターゲットツリーのツリー名になります。

結合操作中に、DSmerge によってソースツリーのツリーオブジェクトのオブジェクトクラスがドメインに変更され、新しいパーティションを作成します。新しいドメインオブジェクトは、新しいパーティションのパーティションルートになります。ソースツリー

のツリーオブジェクトの下位オブジェクトはすべて、ドメインオブジェクトの下に置かれます。

ターゲットツリーの管理者は、作成されたツリーのルートコンテナの権利を持つため、ソースツリーの結合されたルートの権利を持つこととなります。

**注：**権利継承が再計算されて有効になるには、数時間かかる可能性があります。この時間は、ツリーの複雑さ、サイズ、パーティション数によって変化します。

ソースツリーの管理者は、新しく作成されたドメインオブジェクト内でのみ権利を所有します。

**205 ページの 図 31** と **205 ページの 図 32** に、ツリーを特定のコンテナに結合した場合の変化を示します。



図 31 結合前の eDirectory ツリー

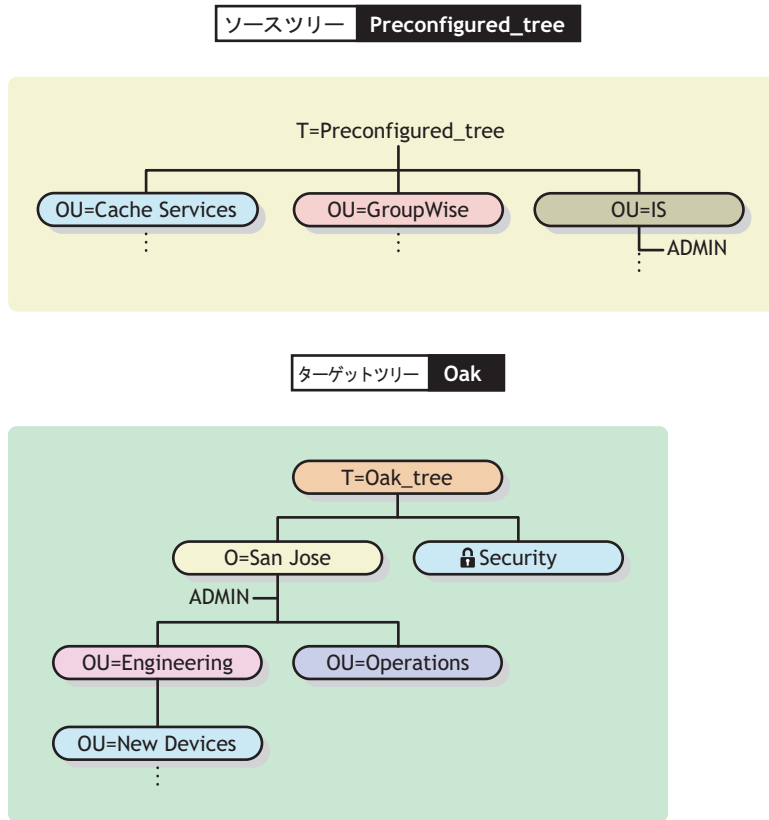
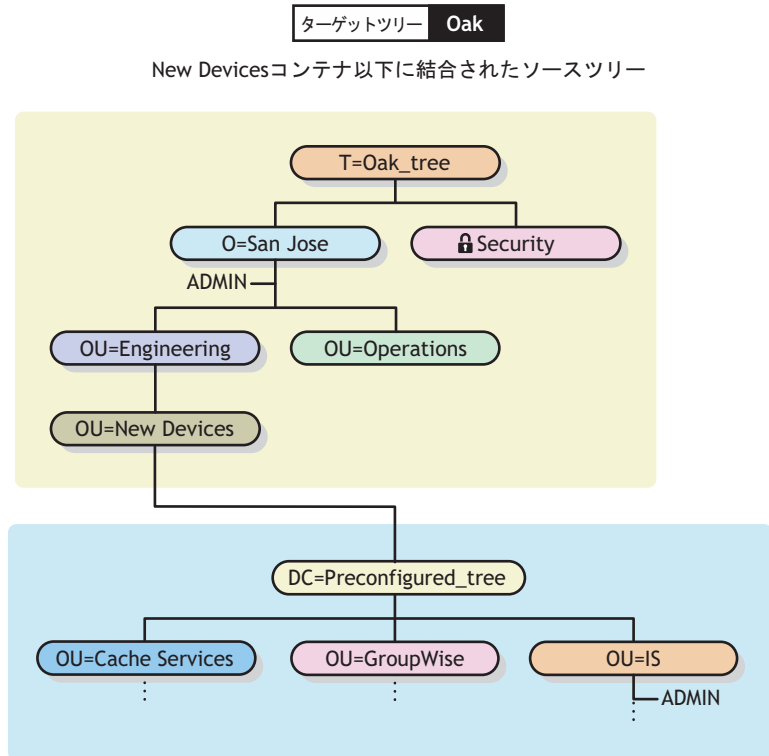


図 32 結合後の eDirectory ツリー



このセクションでは、次の情報について説明します。

- ◆ 206 ページの「コンテキスト名の変更について」
- ◆ 206 ページの「ソースツリーとターゲットツリーを準備する」
- ◆ 208 ページの「結合操作における包含要件」
- ◆ 208 ページの「ソースツリーとターゲットツリーを結合する」

## コンテキスト名の変更について

ソースツリーをターゲットツリーのコンテナに結合すると、ソースツリー内にあるオブジェクトの識別名の後ろに、ソースツリー名と、ソースツリーをマージしたターゲットツリーコンテナの識別名がこの順で追加されます。相対識別名は変わりません。

たとえば、区切り文字としてドットを使用している場合、ソースツリー `Preconfigured_tree` 内にあるオブジェクト `Admin` のタイプ付きの名前は次のようになります。

```
CN=Admin.OU=IS.T=Preconfigured_tree
```

`Preconfigured_tree` を `Oak_tree` の `New Devices` コンテナにマージした後は、`Admin` のタイプ付きの名前は次のようになります。

```
CN=Admin.OU=IS.DC=Preconfigured_tree.OU=Newdevices.  
OU=Engineering.O=Sanjose.T=Oak_tree.
```

**注：**識別名の最大長は 256 文字です。この制限は、1 つのツリーのルートターゲットツリーの下端近くにあるコンテナに結合する場合に特に重要です。

`Oak_tree` の後にある最後のドット (`Oak_tree.`) は、この識別名を構成する最後の要素がツリー名であることを示しています。このドットを省略する場合は、ツリー名も省略します。

## ソースツリーとターゲットツリーを準備する

結合操作を開始する前に、その結合操作の影響を受けるすべてのサーバが安定した状態であることを確認する必要があります。次の表では、結合前のソースツリーとターゲットツリーの準備に関する前提条件について説明します。

前提条件	必要な措置
ソースツリーまたはターゲットツリーのツリーパーティションのレプリカを保持しているすべてのサーバ上で WANMAN がオフになっている。	WANMAN ポリシーを調べて、WAN 通信の制約によるマージ操作への支障がないことを確認します。必要があれば、マージ操作を開始する前に WANMAN をオフにします。
ソースツリー内のサーバが 1 つである。	ソースツリーから、1 つだけ残してすべてのサーバを削除します。
ソースツリーのツリーオブジェクトに別名またはリーフオブジェクトが存在しない。	ソースツリーのツリーオブジェクトに存在する別名またはリーフオブジェクトを削除します。

前提条件	必要な措置
結合コンテナ内に同じ名前がない。	<p>同じ名前がある場合は、ターゲットツリーの結合コンテナ内のオブジェクトまたはソースツリー内のオブジェクト名を変更します。</p> <p>オブジェクト名を変更したくない場合は、コンテナの1つから同じツリー内の別のコンテナにオブジェクトを移動し、空になったコンテナを DSMerge 実行前に削除します。詳細については、<a href="#">3 章 89 ページの、「オブジェクトの管理」</a>を参照してください。</p> <p>同じペアレントオブジェクトの直下でないコンテナオブジェクトの場合は、両ツリーに同じコンテナオブジェクトがあってもかまいません。オブジェクトは、直接のコンテナオブジェクトによって識別されます。</p>
ソースツリーおよびターゲットツリーの eDirectory バージョンが 8.51 SP2a 以降である。	DSMerge は、eDirectory の適切なバージョンを検索します。許容できるバージョンが検出されない場合、DSMerge はエラーを返します。eDirectory の最新バージョンは、 <a href="http://download.novell.com">Novell Download ページ (http://download.novell.com)</a> から取得できます。
ターゲットツリーを結合するコンテナがレプリカを持たないパーティション（単一サーバパーティション）にある。	<p>ターゲットコンテナに複数のレプリカがある場合は、次のうち1つを実行します。</p> <ul style="list-style-type: none"> <li>このコンテナに関連付けられたパーティションをマスターレプリカにし、その他のレプリカを削除します。</li> <li>または、ターゲットツリーの結合コンテナを別のパーティションとして切り離し、レプリカを削除します。</li> </ul> <p>結合が完了した後は、パーティションの関連付けを再構築できます。</p>
ターゲットコンテナを保持するサーバがルートパーティションのレプリカも保持している。	<p>サーバがルートのレプリカを保持していない場合、ディレクトリはターゲットツリーの管理権を確認できないので、結合は失敗し、エラー「-672 アクセス権なし」が表示されます。</p> <p>iManager を使用してルートのレプリカを追加します。詳細については、<a href="#">122 ページの「レプリカを追加する」</a>を参照してください。</p>
ソースツリーおよびターゲットツリーのスキーマが同じである。	<p>DSMerge で [結合] オプションを実行します。出力されるレポートからスキーマに問題があった場合は、ターゲットツリー上で DSRepair を実行してソースツリーからスキーマをインポートします。</p> <p>結合操作によって、自動的にスキーマがターゲットツリーからソースツリーにインポートされます。</p> <p>DSMerge を再実行します。</p>
単一のツリーだけが、ツリーのルートの下位にセキュリティコンテナを保持できる。	ソースツリーおよびターゲットツリーの両方がセキュリティコンテナを保持している場合、 <a href="#">553 ページの付録 A、「NMAS の注意事項」</a> で説明されている手順に従って、一方のコンテナを削除します。

前提条件	必要な措置
ソースツリーのタイムリファレンスが再設定されている。	<p>ソースツリーは、ターゲットツリーのサーバからタイムソースを取得するように、セカンダリサーバとして設定される必要があります。</p> <p>Timesync を再設定するには、『<i>Network Time Management Administration Guide</i>(<i>Network Time Management 管理ガイド</i>)』の「<a href="http://www.novell.com/documentation/japanese/nw65/time_enu/data/abzqzx2.html">Configuring Timesync on Servers (http://www.novell.com/documentation/japanese/nw65/time_enu/data/abzqzx2.html)</a>」を参照してください。</p>


## 結合操作における包含要件

ソースツリーをターゲットツリーのコンテナに結合するには、ターゲットツリーのコンテナでソースツリーの受け入れ準備を整えておく必要があります。ターゲットツリーのコンテナは、クラスドメインのオブジェクトを包含できなければなりません。包含に問題があると、結合操作中に「-611 不正な包含です」というエラーが発生します。

次の表の情報を使用して、DSRepair を実行して包含リストを変更する必要があるかどうかを判断します。


ターゲットツリーのコンテナ要件	<p>ターゲットツリーのコンテナオブジェクトが、包含リスト内のドメインオブジェクトを包含していること。</p> <p>iMonitor で [スキーマ] を使用すると、この状況をチェックできます。包含リストにドメインが包含されていない場合、DSRepair を実行してスキーマを拡張します。</p>
ソースツリーの要件	<p>結合操作によって、ソースツリーのルートがクラスツリールートからクラスドメインに変更されます。ツリーの下位オブジェクトはすべて、スキーマルールに従い、クラスドメインに適切に包含されている必要があります。</p> <p>iMonitor で [スキーマ] を使用すると、この状況をチェックできます。包含リストにドメインが包含されていない場合、DSRepair を実行してスキーマを拡張します。</p>

包含要件が満たされていない場合は、DSRepair を実行してスキーマを修正します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 ユーザ名、パスワード、および操作を実行するサーバのコンテキストを指定し、[次へ] をクリックします。
- 5 [オプションスキーマ拡張機能] をクリックし、[開始] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## ソースツリーとターゲットツリーを結合する

前提条件を満たしていることを確認し、DSMerge を使用して結合を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。

- 2 [eDirectory の保守] > [ツリーの結合] の順にクリックします。
- 3 ソースツリーとして結合を実行するサーバを指定し、[次へ] をクリックします。
- 4 サーバへの認証を行ってから、[次へ] をクリックします。
- 5 ソースツリーの管理者名とパスワード、ターゲットツリー名、およびターゲットツリーの管理者名とパスワードを指定します。
- 6 [開始] をクリックします。

[ツリーの結合] ウィザードのステータスウィンドウが表示され、結合の進行状況が表示されます。結合処理から戻された情報とともに、最後に「完了」メッセージが最後に表示されます。
- 7 [閉じる] をクリックして終了します。

## ツリー名の変更

マージする2つのツリーの名前が同じ場合は、どちらかの名前を変更する必要があります。

ここで名前を変更できるのはソースツリーだけです。ターゲットツリーの名前を変更するには、ターゲットツリー上のいずれかのサーバで Novell iManager の [ツリー名の変更] ウィザードを実行する必要があります。

ツリー名を変更しても、バインダリコンテキストは自動的に変更されません。autoexec.ncfファイルで設定されたバインダリコンテキストセットにもツリー名(例: SET Bindery Context = O=n.test tree\_name) が含まれるため、ツリー名が最近変更されたサーバでは、ツリー名が変更される前のコンテキストは使用されません。

したがって、ツリー名を変更した場合、クライアントワークステーションの環境設定の変更が必要になる可能性があります。Novell Client for DOS/Windows を使用している場合は、net.cfg ファイルの優先ツリーステートメントおよび優先サーバステートメントを確認します。Novell Client for Windows NT/2000 または Novell Client for Windows 95/98 を使用している場合は、クライアントのプロパティページにある優先ツリーステートメントおよび優先サーバステートメントを確認します。

優先サーバが使用されている場合は、ツリーのマージやツリー名の変更を行っても、そのクライアントは名前によってサーバにログインした状態のままなので操作による影響はクライアント側にはありません。優先ツリーが使用されている場合は、ツリーのマージやツリー名の変更を行うと、元のツリー名はなくなります。マージを行った後にはターゲットツリーの名前だけが残ります。優先ツリーの名前を新しいツリー名に変更します。

2つのツリーをマージする場合、ターゲットツリーの名前はマージの結果作成されるツリーの名前としてそのまま残るので、アップデートする必要があるクライアントワークステーションの数を少なくするには、クライアントワークステーションの数が多い方のツリーをターゲットツリーとして指定します。


または、マージの後でツリー名を変更して、最終的なツリー名がクライアントワークステーションの数が多い方のツリーに一致するようにすることもできます。

マージ後のツリーの名前は、元のソースツリーの名前に変更することもできます。その場合は、ターゲットツリーにあるクライアントワークステーションの net.cfg ファイルを更新する必要があります。

次の前提条件の一覧を使用して、リネーム操作の準備ができていないか確認します。

- ❑ ソースツリー上のサーバコンソールへのアクセス権があること、またはこのサーバとの RCONSOLE セッションが確立済みであること
- ❑ ソースツリーのツリーオブジェクトに対するスーパーバイザオブジェクト権があること
- ❑ (オプション) ツリー内のサーバがすべて動作可能であること(動作不能状態のサーバは動作可能になった時点で自動的に更新されます)。

ツリー名を変更するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [ツリー名の変更] の順にクリックします。
- 3 [ツリー名の変更] ウィザードを実行するサーバ(ターゲットツリー内のサーバ)を指定し、[次へ] をクリックします。
- 4 サーバへの認証を行ってから、[次へ] をクリックします。
- 5 新しいツリー名、管理者ユーザ名、およびパスワードを指定します。
- 6 [開始] をクリックします。  
[ツリー名の変更] ウィザードのステータスウィンドウが表示され、リネーム処理の進行状況が表示されます。
- 7 リネーム処理から戻された情報とともに「完了」メッセージが表示されたら、[閉じる] をクリックして終了します。

## eMBox クライアントを使用したツリーのマージ

eMBox(eDirectory Management Toolbox) クライアントはコマンドライン Java クライアントで、これを使用すると DSMerge にリモートアクセスできます。eDirectory の一部としてサーバ上に `emboxclient.jar` ファイルがインストールされます。JVM を実行しているコンピュータならどこからでも実行できます。eMBox クライアントの詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

## DSMerge eMTool を使用する

- 1 非対話式モードで eMBox クライアントを実行するには、コマンドラインに次のコマンドを入力します。

```
java -cp path_to_the_file/emboxclient.jar embox -i
```

(クラスパスにすでに `emboxclient.jar` file が設定されている場合、「`java embox -i`」と入力するだけです)。

次の eMBox クライアントのプロンプトが表示されます。

```
eMBox Client>
```

- 2 DSMerge を実行する(ソースツリーになる)サーバにログインするには、次のコマンドを入力します。

```
login -sserver_name_or_IP_address -pport_number  
-uusername.context -wpassword -n
```

ポート番号は通常 80 または 8008 です。ただし、すでにそのポートを使用している Web サーバが存在する場合は異なります。-n オプションを使用すると、セキュリティ保護されていない接続を開始します。

eMBox クライアントはログインが成功したかどうかを表示します。

- 3 次の構文を使用してマージコマンドを入力します。

```
dsmerge.task options
```

次に例を示します。

「dsmerge.m -uadmin -ptest -TApple -Uadmin -Ptest」は、ターゲットツリー「Apple」を、現在ログインしているソースツリーにマージします。ターゲットツリーのユーザ名は「Admin」、ユーザパスワードは「test」です。また、ソースツリーのユーザ名も「Admin」、ユーザパスワードは「test」です。

「dsmerge.g -uadmin -ptest -TOrange -Uadmin -Ptest -CFruit」は、現在ログインしているソースツリーをターゲットツリー「Orange」内の「Fruit」コンテナに結合します。ソースツリーのユーザ名は「Admin」、ユーザパスワードは「test」です。また、ターゲットツリーのユーザ名も「Admin」、ユーザパスワードは「test」です。

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。

eMBox クライアントは DSMerge 操作が成功したかどうかを表示します。

DSMerge eMTool オプションの詳細については、[479 ページの「eMBox コマンドライククライアントの使用」](#)を参照してください。

- 4 eMBox クライアントからログアウトするには、次のコマンドを入力します。

```
logout
```

- 5 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

## DSMerge eMTool オプション

次の表に、DSMerge eMTool オプションを示します。eMBox クライアントで list -tdsmerge コマンドを使用して、DSMerge オプションの詳細を表示することもできます。詳細については、[483 ページの「eMTool とそのサービスを表示する」](#)を参照してください。

マージ操作	eMBox クライアントのコマンド
ツリーの名前が変更できるかどうかチェックする	dsmerge.pr -uUser -pUser_password -nNew_tree_name
ツリー名を変更する	dsmerge.r -uUser -pUser_password -nNew_tree_name
2 つのツリーがマージできるかどうかチェックする	dsmerge.pm -uSource_tree_user -pSource_tree_user_password -TTarget_tree_name -UTarget_tree_user -PTarget_tree_password
2 つのツリーをマージする	dsmerge.m -uSource_tree_user -pSource_tree_user_password -TTarget_tree_name -UTarget_tree_user -PTarget_tree_password
ソースツリーがターゲットツリーのコンテナに結合できるかどうかチェックする	dsmerge.pg -uSource_tree_user -pSource_tree_user_password -TTarget_tree_name -UTarget_tree_user -PTarget_tree_password -CTarget_tree_container
ソースツリーをターゲットツリーのコンテナに結合する	dsmerge.g -uSource_tree_user -pSource_tree_user_password -TTarget_tree_name -UTarget_tree_user -PTarget_tree_password -CTarget_tree_container
実行中の DSMerge 操作をキャンセルする	cancel





# 9

## Novell eDirectory データベースの修復

修復ユーティリティを使用すると、Novell® eDirectory™ ツリーのデータベースを保守および修復することができます。このユーティリティでは、次の操作を実行できます。

- ◆ 不正なレコード、スキーマの不一致、不正なサーバアドレス、外部参照など、eDirectory の問題の修正
- ◆ eDirectory スキーマへの詳細な変更
- ◆ データベースの終了やユーザの介入を伴わない、自動的なデータベースの構造チェック
- ◆ データベースのオペレーショナルインデックスの確認
- ◆ 空のレコードを破棄することによる空き領域の増量
- ◆ ローカルデータベースの修復
- ◆ レプリカ、レプリカリングおよびサーバオブジェクトの修復
- ◆ 同期エラーに対する、各ローカルパーティションの各サーバの分析
- ◆ ローカルデータベース内のオブジェクトの検出と同期

すべての eDirectory データベースの問題が致命的であるというわけではなく、eDirectory によって処理を続行できる問題もあります。ただしデータベースが破損すると、ローカルデータベースを開くことができないことを知らせるメッセージがコンソールに表示されます。この場合、修復を実行するか Novell サポートに連絡します。

eDirectory に問題が発生したり Novell サポートから指示がない限り、修復操作を実行することはお勧めできません。ただし、修復ユーティリティおよび他の Novell ユーティリティ (Novell iMonitor など) の診断機能を使用することはお勧めします。詳細については、7 章 171 ページの、「Novell iMonitor 2.1 の使用」を参照してください。

Novell iManager には次の修復ウィザードが含まれています。

ウィザード	説明
基本修復ウィザード	無人修復、ローカルデータベースの修復、または単一オブジェクトの修復を実行できます。また、外部参照をチェックして不明なリーフオブジェクトを削除できます。
ログファイルウィザード	修復ログファイルを開いて、ログファイルオプションを設定できます。
iMonitor による修復	iMonitor を開いて、このプログラムの修復オプションを使用できます。
レプリカ修復ウィザード	すべてまたは選択したレプリカの修復、タイムスタンプの修復と新しいエポックの宣言、現在のサーバを新しいマスターレプリカに設定、および必要に応じて選択したレプリカの削除などを実行できます。

ウィザード	説明
レプリカリングの修復ウィザード	すべてまたは選択したレプリカリングの修復、リング内のすべてのサーバにすべてのオブジェクトを送信、選択したレプリカでマスタレプリカのすべてのオブジェクトを受信、および必要に応じてレプリカリングから現在のサーバの削除などを実行できます。
スキーマの保守ウィザード	ツリーからスキーマを要求、ローカルスキーマのリセット、新しいスキーマエポックの宣言、オプションスキーマ拡張機能の実行、リモートスキーマのインポート、および Post NetWare® 5 スキーマの更新などを実行できます。
サーバの修復ウィザード	すべてのネットワークアドレスの修復、およびサーバのネットワークアドレスのみの修復ができます。
同期修復ウィザード	現在のサーバで選択したレプリカの同期、現在のサーバの同期ステータスのレポート、すべてのサーバの同期ステータスのレポート、時刻同期の実行、および即時同期のスケジュールが実行できます。

ウィザードは、次のような操作のときに役立ちます。

- ◆ [214 ページの「基本修復操作の実行」](#)
- ◆ [219 ページの「修復ログファイルの表示と設定」](#)
- ◆ [219 ページの「Novell iMonitor での修復の実行」](#)
- ◆ [220 ページの「レプリカの修復」](#)
- ◆ [223 ページの「レプリカリングを修復する」](#)
- ◆ [225 ページの「スキーマの保守」](#)
- ◆ [229 ページの「サーバのネットワークアドレスの修復」](#)
- ◆ [230 ページの「同期化操作の実行」](#)
- ◆ [232 ページの「DSRepair のカスタムオプション」](#)
- ◆ [236 ページの「eMBox クライアントを使用したデータベースの修復」](#)

## 基本修復操作の実行

基本修復ウィザードでは、無人修復、ローカルデータベースの修復、または単一オブジェクトの修復を実行できます。また、外部参照をチェックして不明なリーフオブジェクトを削除できます。

- ◆ [215 ページの「無人修復を実行する」](#)
- ◆ [217 ページの「ローカルデータベースの修復の実行」](#)
- ◆ [217 ページの「外部参照のチェック」](#)
- ◆ [218 ページの「単一オブジェクトの修復」](#)
- ◆ [218 ページの「不明なリーフオブジェクトの削除」](#)

## 無人修復を実行する

無人修復では、指定されたサーバの eDirectory データベースファイルに致命的な eDirectory エラーがないかチェックし、修復します。このオプションは、実行されるたびに 8 つの主要な操作を実行します。これらの操作には、管理者が関与する必要はありません。これらの操作の中には、実行中にデータベースをロックするものがあります。無人修復では、ローカルデータベースファイルのセットが一時的に作成され、修復操作はこれらのファイルに対して実行されます。つまり、重大な問題が発生したとしても、オリジナルのファイルは無事です。

ローカルデータベースの修復で使用される対話式オプションを熟知していない場合は、この修復手段をお勧めします。無人修復を実行すると、データベースファイルが現在使用している 2 倍の容量の空きディスクが必要になる場合があります。詳細については、[217 ページの「ローカルデータベースの修復の実行」](#)を参照してください。


eDirectory が使用するオペレーショナルインデックスの再構築は、ローカルデータベースがロックされているときのみ可能です。

次の表に、無人修復の実行中に行われる操作について示します。

操作	データベースのロック	説明
データベース構造およびインデックスのチェック	する	データベースレコードとインデックスの構造および形式を調べます。eDirectory 環境のデータベースレベルで構造的な破損が組み込まれていないことを確認します。
データベース全体の再構築	する	構造チェックおよびインデックスチェック中に見つかったエラーを解決します。正しいデータ構造を復元し、eDirectory データベースおよびインデックスファイルを再作成します。
ツリー構造のチェックの実行	する	データベースレコード間のリンクを検証し、各チャイルドレコードに対して有効なペアレントレコードがあることを確認します。これは、データベースの整合性を確認するのに役立ちます。無効なレコードにはマークがつけられ、eDirectory レプリカ同期処理の実行中に別のパーティションレプリカから復元できます。
すべてのローカルレプリカを修復	する	各オブジェクトと属性をスキーマ定義でチェックし、eDirectory データベースの不整合を解決します。ここでは、内部データ構造の形式もすべてチェックします。  この操作では、データベースから無効なレコードを削除することにより、ツリー構造のチェック中に見つかった不整合も解決します。この結果、無効なレコードにリンクされているすべてのチャイルドレコードは、すべてオーファンとしてマークされます。これらのオーファンレコードは失われませんが、この処理によって、データベースの再構築中に多数のエラーが発生する可能性があります。これは正常な反応で、孤立したオブジェクトはレプリカ同期の過程で自動的に再編成されます。

操作	データベース のロック	説明
ローカル参照をチェック	する	ローカル参照は、このファイルサーバ上の eDirectory データベース内で管理されているオブジェクトへのポインタです。この操作では、内部データベースのポインタを評価し、正しい eDirectory オブジェクトを指していることを確認します。無効な参照が見つかった場合は修正されます。オブジェクト間に存在する関係の数によって異なりますが、この操作は完了までにかかり時間がかかる場合があります。
ネットワークアドレスの修復	しない	eDirectory 内で保存しているサーバのネットワークアドレスを、ローカル SAP、SLP、または DNS テーブルで維持されている値でチェックし、eDirectory が現在も正確な情報を保持していることを確認します。矛盾が見つかった場合、eDirectory は正しい情報で更新されます。
ストリームシンタックスファイルの確認	する	ログインスクリプトなどのストリームシンタックスファイルは、eDirectory データベースの特殊領域に保存されます。この操作では、各ストリームシンタックスファイルが有効な eDirectory オブジェクトに関連付けられているかどうかをチェックします。関連付けられていないストリームシンタックスファイルは削除され、そのファイルを参照している属性はパージされます。
メールディレクトリを確認する (NetWare のみ)	する	デフォルトでは、eDirectory はメールディレクトリを NetWare <sup>®</sup> サーバの sys:mail ディレクトリ内に作成します。これは以前のバインダリユーザをサポートするためです。バインダリユーザのログインスクリプトは、ユーザのメールディレクトリに保存されます。この操作では、各メールディレクトリが有効な eDirectory ユーザオブジェクトに関連付けられているかどうかをチェックします。関連付けられていない場合、メールディレクトリは削除されます。
ポリシーオブジェクトとトラスティのチェック (NetWare のみ)	しない	NetWare サーバ上の各ポリシーが、eDirectory 内のポリシーオブジェクトに関連付けられていることを確認します。関連付けられていない場合、サーバが存在するコンテキストを検索し、ポリシーオブジェクトが存在しているかどうか確認します。ポリシーオブジェクトが存在しない場合は、ポリシーオブジェクトを作成します。  ポリシー情報を検証後、トラスティ ID のリストが検証されます。eDirectory 内のすべてのオブジェクトには固有のトラスティ ID があります。この ID は、NetWare ポリユームなど eDirectory ツリー内の他のオブジェクトに権利を与える際に使用されます。この作業では、ポリシーリストにある各トラスティ ID が有効な eDirectory オブジェクトであることを確認します。そうでない場合、トラスティ ID はポリシーリストから削除されます。

無人修復を実行するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [基本修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。

- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [標準修復] をクリックし、[開始] をクリックします。
- 6 表示される指示に従って、操作を完了します。


## ローカルデータベースの修復の実行

eDirectory でオープンおよびアクセスできるように、この修復操作を使用してローカルデータベースの矛盾を解決します。

ローカルデータベースの修復は、一時ファイルセットに対して実行するように指定することもできます。一時ファイルセットを指定しなかった場合、修復操作はアクティブなデータベースに対して実行されます。

一時データベースファイルセットに対して修復操作を実行する場合は、この操作中はデータベースを閉じておく必要があります。操作対象を一時ファイルセットにした場合、修復結果を反映する前に、その確認を求めるメッセージが表示されます。それ以外の場合、修復結果は即座に反映されます。


修復操作が終了すると、その修復操作のログを表示して、修復を完了させるのにさらに必要な操作があるかどうかを確認できます。詳細については、[219 ページの「修復ログファイルの表示と設定」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [基本修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [ローカルデータベースの修復] をクリックし、[次へ] をクリックします。
- 6 ローカル修復を実行するオプションを指定し、[開始] をクリックします。
- 7 表示される指示に従って、操作を完了します。

## 外部参照のチェック

この修復操作は、各外部参照オブジェクトをチェックして、そのオブジェクトを含むレプリカがあるかどうかを調べます。オブジェクトのあるパーティションのレプリカが含まれているすべてのサーバにアクセスできない場合、オブジェクトは見つけれられません。オブジェクトが見つからない場合、警告が表示されます。

この操作では破損情報も表示されます。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [基本修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [外部参照のチェック] をクリックし、[開始] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 単一オブジェクトの修復

この修復操作は、eDirectory がデータへアクセスするのを妨げるような、eDirectory オブジェクトの不整合を解決します。この操作は、ユーザ作成のパーティションおよび外部参照パーティションでのみ有効です。

この操作は、アクティブなデータベースファイルに対して実行されます。破損が物理的なレベルの場合は、まず物理チェックおよび構造チェックを実行してから単一オブジェクトの修復を行います。

修復操作時点の eDirectory データベースのバックアップコピーを保持していることを確認します。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [基本修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [単一オブジェクトの修復] をクリックし、[開始] をクリックします。
- 6 修復するオブジェクトを指定し、[次へ] をクリックします。
- 7 表示される指示に従って、操作を完了します。

## 不明なリーフオブジェクトの削除

オブジェクトに必須プロパティがない場合、あるいはその他に無効な点がある場合(プロパティがオブジェクトタイプの最低要件を満たしていない場合)、修復によって一貫性のないオブジェクトが不明なオブジェクトに変更されます。不明なオブジェクトは実際のオブジェクトであり、eDirectory 側では既知のオブジェクトです。不明なオブジェクトになっているのは、オブジェクトクラスの検証が不完全なためです。疑問符アイコンで表示される不明なオブジェクトは削除できますが、簡単に元のオブジェクトタイプに戻すことはできません。

この修復操作では、ローカル eDirectory データベースのオブジェクトのうち、オブジェクトクラスが不明で、従属オブジェクトを維持していないオブジェクトをすべて削除します。削除は eDirectory ツリーの他のレプリカと同時に後で行われます。

**重要:** この操作の意味を完全に理解しているか Novell サポートから実行の指示がない限り、この操作は実行しないでください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [基本修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [不明なリーフオブジェクトの削除] をクリックし、[開始] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 修復ログファイルの表示と設定

修復ログファイルには、ローカルパーティションとサーバに関する詳細情報が含まれます。この情報はデータベースの破損を診断するのに役立ちます。ログファイルウィザードでは、修復ログファイルを開いてログファイルオプションを設定できます。


このセクションでは、次の操作について説明します。

- ◆ [219 ページの「ログファイルを開く」](#)
- ◆ [219 ページの「ログファイルオプションを設定する」](#)

### ログファイルを開く


この操作を行って、修復ログファイルを表示します。ログファイルのデフォルト名は dsrepair.log です。修復によって実行される操作の結果がログファイルに書き込まれます。

ログファイル操作のオン/オフ切り替え、名前の変更、およびログファイルの削除またはリセットなどができます。詳細については、[219 ページの「ログファイルオプションを設定する」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [ログファイル] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [ログファイルを開く] をクリックし、[開く] をクリックします。
- 6 表示される指示に従って、操作を完了します。

### ログファイルオプションを設定する

この操作を行って、修復ログファイルを管理します。ログファイルのオン/オフ切り替え、ログファイルの削除、追加、ファイル名の変更などが行えます。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [ログファイル] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [ログファイルオプション] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## Novell iMonitor での修復の実行

Novell iManager の [iMonitor による修復] オプションを使用して、修復機能にアクセスできます。iMonitor の [修復] ページでは、問題を表示したり、eDirectory データベースのバックアップやクリーンアップを実行できます。

iMonitor では、DSRepair 機能はサーバ限定の機能です。つまり、この機能は iMonitor が実行されているローカルサーバでのみ使用できます。他のサーバで実行されている DS トレース機能にアクセスするには、そのサーバで実行されている iMonitor に切り替える必要があります。

[DS Repair] ページにアクセスするには、アクセスしようとするサーバの管理者またはコンソールオペレータと同等の権利が必要です。つまり、このページの情報にアクセスするには、まずログインして認証情報のチェックを受ける必要があります。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [iMonitor による修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[OK] をクリックします。  
手動で iMonitor を開いて修復オプションを実行するには、[iMonitor を実行して iMonitor から修復にアクセスする] をクリックした後で [OK] をクリックします。
- 4 アクセスするサーバのユーザ名、コンテキスト、およびパスワードを指定してから [OK] をクリックし、[iMonitor 修復] ページを開きます。
- 5 修復オプションを指定し、[修復の開始] をクリックします。

iMonitor で利用可能な修復機能の詳細については、187 ページの「DSRepair 情報の表示」を参照してください。

## レプリカの修復

レプリカの修復操作では、レプリカの各オブジェクトとスキーマとの整合性が保たれているかどうか、オブジェクトの各属性とスキーマとの整合性が保たれているかどうかをチェックし、属性のシンタックスに従ってデータをチェックします。レプリカに関連する他の内部データ構造もチェックされます。


レプリカの修復ウィザードを使用して、次の操作を実行します。

- ◆ 220 ページの「すべてのレプリカを修復する」
- ◆ 221 ページの「選択したレプリカを修復する」
- ◆ 221 ページの「タイムスタンプを修復する」
- ◆ 222 ページの「このサーバを新しいマスタレプリカに設定する」
- ◆ 223 ページの「選択したレプリカを削除する」

## すべてのレプリカを修復する

この操作では、レプリカテーブルに表示されたすべてのレプリカを修復します。

30 分前までにローカル eDirectory データベースの修復操作を行っていない場合、この操作を実行する前にローカルデータベースを修復してください。詳細については、217 ページの「ローカルデータベースの修復の実行」を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。




- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [すべてのレプリカの修復] をクリックし、[開始] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 選択したレプリカを修復する

この操作では、レプリカビューに表示されているレプリカのうち、選択したレプリカのみ修復します。

30 分前までにローカル eDirectory データベースの修復操作を行っていない場合、この操作を実行する前にローカルデータベースを修復してください。詳細については、[217 ページの「ローカルデータベースの修復の実行」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [選択したレプリカを修復する] をクリックし、[次へ] をクリックします。
- 6 修復するレプリカを指定し、[開始] をクリックします。
- 7 表示される指示に従って、操作を完了します。

## タイムスタンプを修復する

**注：**この操作を行う前に、同期修復ウィザードを使用して、レプリカリング内のすべてのサーバが正しく通信していることを確認します。詳細については、[230 ページの「同期化操作の実行」](#)を参照してください。

この操作では、選択したパーティションのレプリカをすべて最新版に更新するために、マスタレプリカの新しい参照ポイントを指定します。

この操作は、常にパーティションのマスタレプリカ上で実行されます。マスタレプリカは、このサーバのローカルレプリカである必要はありません。


オブジェクトが作成または変更されるとタイムスタンプが設定されますが、これは一意でなければなりません。マスタレプリカのすべてのタイムスタンプが検査されます。タイムスタンプが現在のネットワーク時間より遅れている場合、新しいタイムスタンプに置き換えられます。タイムスタンプが最新であれば、新しいタイムスタンプは発行されません。すべてのタイムスタンプの時刻が一致すると、新規エポックが宣言されます。

この操作は、レプリカのオブジェクト間、またはオブジェクトのプロパティ間で矛盾が生じている場合に使用します。たとえば、ログインスクリプトを更新したのにログイン時に古いログインスクリプトが表示される場合は、レプリカ間で正しく同期が取られているかどうかを確認してください。将来のタイムスタンプと現在の時刻の時間差が 1 分以内であれば、最終的に eDirectory 自体がその状況を修正します。新規エポックの宣言は非常に費用のかかる操作であり、定期的な使用はお勧めしていません。

Novell eDirectory はデータベースとして厳密な整合性はとられていません。したがってそのレプリカ同期の確認には 5 ~ 10 分かかることがあります。この操作を行うと、次の状態になります。

- ◆ 新規エポックがマスタレプリカで宣言され、その影響がマスタレプリカのすべてのオブジェクトに及ぶ可能性があります。
- ◆ すべてのタイムスタンプが調べられ、必要に応じて修復されます。
- ◆ レプリカ間の同期が取られるまで、日付の古いタイムスタンプ(エポック)を保持するレプリカからの更新内容は受け付けられません。
- ◆ レプリカは、マスタレプリカまたは新規エポックを受信済みの他のレプリカのすべてのオブジェクトのコピーを受け取ります。
- ◆ このレプリカは、マスタレプリカと同じエポックになります。
- ◆ 以前のエポックからの変更内容は失われます。
- ◆ マスタレプリカが現在のサーバに存在する必要はありませんが、この修復操作を実行するにはマスタレプリカに対するスーパーバイザ権が必要です。
- ◆ そのほかのレプリカは新しい状態になります。


タイムスタンプを修復して新規エポックを宣言するには、次の操作を行います。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [タイムスタンプを修復して新しいエポックを宣言する] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## このサーバを新しいマスタレプリカに設定する

この操作では、選択したパーティションのローカルレプリカをマスタレプリカとして設定します。元のマスタレプリカを損失した場合には、この操作で新しいマスタレプリカを設定できます。マスタレプリカがあるサーバでハードディスク障害が発生すると、そのマスタレプリカが失われることがあります。その場合は、マスタレプリカを変更する必要があります。


Novell iManager で使用可能な通常のパーティション操作を実行するためには、このオプションを使用しないでください。詳細については、[5章 117 ページの、「パーティションおよびレプリカの管理」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカの修復] の順にクリックします。
- 3 新しいマスタレプリカに指定するサーバを指定し、[次へ] をクリックします。
- 4 ユーザ名、パスワード、およびコンテキストを指定してサーバへの認証を行い、[次へ] をクリックします。
- 5 [このサーバを新しいマスタレプリカに設定] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 選択したレプリカを削除する

この操作では、選択したレプリカをこのサーバから削除します。レプリカは削除されるか、サブオーディネートリファレンスに変更されます。

Novell iManager で使用可能な通常のパーティション操作を実行するためには、このオプションを使用しないでください。詳細については、[5 章 117 ページの「パーティションおよびレプリカの管理」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカの修復] の順にクリックします。
- 3 削除するレプリカを含むサーバを指定し、[次へ] をクリックします。
- 4 ユーザ名、パスワード、およびコンテキストを指定してサーバへの認証を行い、[次へ] をクリックします。
- 5 [選択したレプリカを削除する] をクリックし、[次へ] をクリックします。
- 6 削除するレプリカを指定し、[次へ] をクリックします。
- 7 表示される指示に従って、操作を完了します。

## レプリカリングを修復する

レプリカリングの修復操作では、レプリカを保持する各サーバのレプリカリング情報をチェックし、リモート ID 情報を検証します。


レプリカリング修復ウィザードを使用して、次の操作を実行します。

- ◆ [223 ページの「すべてのレプリカリングを修復する」](#)
- ◆ [224 ページの「選択したレプリカリングを修復する」](#)
- ◆ [224 ページの「リング内のすべてのサーバにすべてのオブジェクトを送信する」](#)
- ◆ [225 ページの「マスタから選択したレプリカへすべてのオブジェクトを受信する」](#)
- ◆ [225 ページの「レプリカリングからこのサーバを削除する」](#)

## すべてのレプリカリングを修復する

この操作では、レプリカビューに表示されたすべてのレプリカのレプリカリングを修復します。

30 分前までにローカル eDirectory データベースの修復操作を行っていない場合、この操作を実行する前にローカルデータベースを修復してください。詳細については、[217 ページの「ローカルデータベースの修復の実行」](#)を参照してください。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカリングの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。

- 5 [Repair All Replica Rings(すべてのレプリカリングを修復)]をクリックし、[次へ]をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 選択したレプリカリングを修復する

この操作では、レプリカテーブルで選択したレプリカのレプリカリングを修復します。

30分前までにローカル eDirectory データベースの修復操作を行っていない場合、この操作を実行する前にローカルデータベースを修復してください。詳細については、[217 ページの「ローカルデータベースの修復の実行」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカリングの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [Repair the Selected Replica Ring(選択されたレプリカリングを修復)] をクリックし、[次へ] をクリックします。
- 6 修復するレプリカを指定し、[次へ] をクリックします。
- 7 表示される指示に従って、操作を完了します。


## リング内のすべてのサーバにすべてのオブジェクトを送信する

この操作では、レプリカリング内で選択したサーバから、選択したパーティションのレプリカを含む他のすべてのサーバに、すべてのオブジェクトを送信します。

レプリカリング内で選択したサーバ上の選択したパーティションのレプリカが、レプリカリング内の他のすべてのサーバと同期していることを確かめるには、この操作を行います。該当するパーティションのサブオーディネートリファレンスレプリカのみを保持するサーバではこの操作は実行できません。

選択したサーバに保持されているレプリカとまだ同期していない他のレプリカに加えた変更は失われます。この操作を実行する前に、同期ステータスを確認してください。

**重要**：この操作はレプリカ内のオブジェクトを再作成するため、ネットワークトラフィックの量が大幅に増加する可能性があります。これは診断操作ではありません。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカリングの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 サーバのユーザ名、パスワード、およびコンテキストを指定して、[次へ] をクリックします。
- 5 [リング内の各サーバにすべてのオブジェクトを送信する] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## マスタから選択したレプリカへすべてのオブジェクトを受信する

この操作では、選択したサーバ上のレプリカで、マスタレプリカのすべてのオブジェクトを受信します。

レプリカリング内で選択したサーバ上の選択したパーティションのレプリカが、マスタレプリカと同期していることを確かめるには、この操作を行います。この操作は、マスタレプリカがあるサーバでは実行できません。


**重要**：この操作を行うと、ネットワークトラフィックの量が大幅に増加します。この操作を要求することにより、現在のレプリカは、サーバに新しいレプリカがあるかのように動作します。さらに、レプリカの状態は新規になります。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカリングの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 サーバのユーザ名、パスワード、およびコンテキストを指定して、[次へ] をクリックします。
- 5 [マスタから選択したレプリカにすべてのオブジェクトを受信する] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## レプリカリングからこのサーバを削除する

この操作では、現在のサーバに保存されているレプリカのうち、選択したレプリカから特定のサーバを削除します。

**警告**：この操作を誤用すると、eDirectory データベースで致命的な破損が生じることがあります。Novell サポート担当者からの指示がない限り、この操作は実行しないでください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [レプリカリングの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 サーバのユーザ名、パスワード、およびコンテキストを指定して、[次へ] をクリックします。
- 5 [レプリカリングからこのサーバを削除] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## スキーマの保守

スキーマとはオブジェクト属性のルールおよび定義を体系化したもので、これにより、各オブジェクトの内容と形式が構築され、データベース内でのオブジェクト間の関係が確立されます。

スキーマの保守ウィザードには、eDirectory サーバのスキーマを [Root] のマスタに合わせるために必要なスキーマ操作がいくつか用意されています。ただし、これらの操作は必要などきだけ使用してください。スキーマは、ローカル修復操作および無人修復操作によってすでに検査されています。

eDirectory スキーマの詳細については、4 章 105 ページの、「スキーマの管理」を参照してください。


スキーマの保守ウィザードを使用して、次の操作を実行します。

- ◆ 226 ページの「ツリーからスキーマを要求する」
- ◆ 226 ページの「ローカルスキーマをリセットする」
- ◆ 227 ページの「Post-NetWare 5 スキーマの更新を実行する」
- ◆ 227 ページの「オプションスキーマ拡張機能を実行する」
- ◆ 228 ページの「リモートスキーマをインポートする」
- ◆ 228 ページの「スキーマの新規エポックを宣言する」

## ツリーからスキーマを要求する

この操作を実行すると、ツリーのルートのマスタレプリカが自身のスキーマをこのサーバのスキーマに同期させます。[Root] のマスタレプリカからこのサーバのスキーマに対して行われた変更は、24 時間以内に伝達されます。


**重要:** すべてのサーバがマスタレプリカのスキーマを要求すると、ネットワークトラフィックが増加します。そのため、このオプションの使用には細心の注意を払うようにしてください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [ツリーからスキーマを要求] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## ローカルスキーマをリセットする

この操作は、ローカルスキーマのタイムスタンプをクリアし、着信スキーマの同期を要求するスキーマリセット機能呼び出します。

[Root] パーティションのマスタレプリカから実行した場合、この操作はできません。ツリー内のすべてのサーバが同時にリセットされるわけではありません。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [ローカルスキーマのリセット] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## Post-NetWare 5 スキーマの更新を実行する

この操作では、Post-NetWare 5 DS の変更に伴い、互換性を確保するためにスキーマを拡張および変更します。

現在の eDirectory バージョンによっては、新しいバージョンにアップデートするときに、このオプションが必要な場合があります。このオプションが必要かどうかは、アップグレードする新しい eDirectory のリリースノートを参照してください。

この操作には、このサーバが [Root] パーティションのレプリカ ([Root] のマスタレプリカを推奨) を持ち、レプリカが使用可能な状態であることが必要になります。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [Post NetWare 5 スキーマの更新] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。


## オプションスキーマ拡張機能を実行する

この操作では、包含など拡張機能のためにスキーマを拡張および変更します。

この操作では、このサーバに [Root] パーティションのレプリカが含まれ、そのレプリカが使用可能な状態であることが必要になります。また、ツリー内のどの NetWare 4 サーバにも次のいずれかのバージョンの ds.nlm が搭載されている必要があります。

サーバ	バージョン
4.10	ds.nlm v5.17 以降
4.11 / 4.2	ds.nlm v6.01 以降

以前のバージョンの eDirectory では、このオプションで加えた変更の同期ができません。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [オプションスキーマ拡張機能] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## リモートスキーマをインポートする

この操作では、現在のツリーのスキーマに追加したいスキーマが含まれている eDirectory ツリーを選択します。

ツリーを選択すると、[Root] パーティションのマスタレプリカを保持するサーバに接続されます。現在のツリー上にあるスキーマの拡張には、そのサーバのスキーマが使用されます。

2 つのツリーをマージするには、一方のツリーからもう一方のツリーへ繰り返しスキーマをインポートする必要があります。詳細については、[8 章 197 ページの、「Novell eDirectory ツリーのマージ」](#)を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [リモートスキーマのインポート] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## スキーマの新規エポックを宣言する

エポックとは、基準点として任意に選択される瞬間のことです。時代や新規バージョンと同義です。エポックは、レプリカの同期を制御します。新規に宣言されると、マスタレプリカで機能します。他のレプリカは、新規エポックを持つレプリカに更新情報を送ることはできませんが、そのレプリカと完全に同期するまでは更新情報を受け取ります。


指定したパーティションの他のレプリカが更新済みレプリカと同期された場合、つまり各レプリカのエポックが同じになると、双方向の同期が再び許可されます。

スキーマの新規エポックを宣言すると、[Root] パーティションのマスタレプリカを保持するサーバに接続され、スキーマレコード上の不正なタイムスタンプが修復されます。次にスキーマの新規エポックがそのサーバで宣言され、ツリー全体に影響を与えます。

他のすべてのサーバは、修復されたタイムスタンプを保持する新しいスキーマのコピーを受け取ります。

受け取る側のサーバが新規エポック内に存在しなかったスキーマを含む場合は、古いスキーマを使用するオブジェクトおよび属性が「不明」オブジェクトクラスまたは属性に変更されます。

**重要:** Novell サポートからの指示がない限り、この操作は実行しないでください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [スキーマの保守] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [新規エポックの宣言] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。



## サーバのネットワークアドレスの修復

サーバの修復ウィザードでは、レプリカリングのすべてのサーバのネットワークアドレスとローカルデータベースのサーバオブジェクトを修復します。さらに、レプリカリングの指定したサーバのネットワークアドレスとローカルデータベースのサーバオブジェクトも修復できます。

サーバの修復ウィザードを使用して、次の操作を実行します。


- ◆ [229 ページの「すべてのネットワークアドレスを修復する」](#)
- ◆ [229 ページの「サーバのネットワークアドレスを修復する」](#)

### すべてのネットワークアドレスを修復する

この操作では、ローカル eDirectory データベース内で、すべてのサーバのネットワークアドレスをチェックします。使用できるトランスポートプロトコルに応じて、SAP テーブル、SLP ディレクトリエージェント、および DNS ローカルまたはリモート情報でサーバ名を検索します。

その後、eDirectory サーバオブジェクトのネットワークアドレス属性、およびすべてのパーティション [Root] オブジェクトの各レプリカ属性のアドレスレコードと、各アドレスが比較されます。アドレスが異なる場合は、同じになるように更新されます。


SAP テーブル、ローカル/リモート DNS 情報、または SLP ディレクトリエージェントにサーバのネットワークアドレスが見つからなければ、修復は行われません。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [サーバの修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [すべてのネットワークアドレスの修復] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

### サーバのネットワークアドレスを修復する

この操作では、ローカル eDirectory データベース内で、選択したサーバのネットワークアドレスをチェックします。現在バインドされているトランスポートプロトコルに応じて、ローカル SAP テーブル、SLP ディレクトリエージェント、およびローカルまたはリモートの DNS 情報でサーバ名を検索します。ネットワークアドレスが見つければ、そのアドレスを eDirectory サーバオブジェクトのネットワークアドレス属性の値、およびすべてのパーティション [Root] オブジェクトのレプリカ属性のアドレスレコードと照合します。アドレスが異なる場合は、同じになるように更新されます。

SAP テーブル、SLP ディレクトリエージェント、またはローカル/リモート DNS 情報にサーバのネットワークアドレスが見つからなければ、修復は行われません。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [サーバの修復] の順にクリックします。

- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [このサーバのネットワークアドレスの修復] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 問題

Novell SLP はオプションのパッケージです。認証機能は、Novell SLP パッケージの一部として実装されているわけではありません。

eDirectory は現在 OpenSLP と相互運用が可能で、OpenSLP の認証機能が使用されます。

## 同期化操作の実行

同期修復ウィザードでは、現在のサーバで選択したレプリカの同期、現在のサーバの同期ステータスのレポート、すべてのサーバの同期ステータスのレポート、時刻同期の実行、および即時同期のスケジュールができます。

同期修復ウィザードを使用して、次の操作を実行します。


- ◆ 230 ページの「選択したレプリカをこのサーバで同期する」
- ◆ 231 ページの「このサーバの同期ステータスをレポートする」
- ◆ 231 ページの「すべてのサーバの同期ステータスをレポートする」
- ◆ 231 ページの「時刻同期を実行する」
- ◆ 232 ページの「即時同期をスケジュールリングする」

## 選択したレプリカをこのサーバで同期する

この操作では、選択したパーティションのレプリカを持つすべてのサーバ上で、完全な同期ステータスを確保します。

このオプションを使用して、パーティションの状態を確認できます。そのパーティションのレプリカを持つサーバがすべて正常に同期していれば、そのパーティションは異常なしと見なされます。レプリカリング内の各サーバが接続されると、次に、接続された各サーバは、レプリカリング内の他のすべてのサーバに対して、即時に同期を実行します。

サーバは、サーバ自体には同期されません。したがって、現在のサーバが所有するレプリカのステータスは「ホスト」として表示されます。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [同期修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [このサーバ上で選択したレプリカを同期する] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## このサーバの同期ステータスをレポートする

この操作では、現在のサーバ上にレプリカを持つすべてのパーティションのレプリカ同期ステータスをレポートします。

この操作では、パーティションのレプリカを保持する各サーバのレプリカの [Root] オブジェクトから同期ステータス属性を読み込みます。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。

12 時間以内に同期が完了していない場合は、警告メッセージが表示されます。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [同期修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [Report the Sync Status on This Server( このサーバの同期ステータスのレポート )] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## すべてのサーバの同期ステータスをレポートする

この操作では、現在のサーバ上にレプリカを持つすべてのパーティションのレプリカ同期ステータスを確保します。

この操作では、パーティションのレプリカを保持する各サーバのレプリカの [Root] オブジェクトから同期ステータス属性を読み込みます。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。

12 時間以内に同期が完了していない場合は、警告メッセージが表示されます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [同期修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [すべてのサーバの同期ステータスのレポート] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 時刻同期を実行する

この操作では、ローカル eDirectory データベースに登録されているすべてのサーバに接続し、各サーバの eDirectory と時刻同期ステータスに関する情報を要求します。

各サーバ上で実行している eDirectory のバージョンが [DS バージョン] フィールドに表示されます。

サーバに何のレプリカも含まれていない場合は、[レプリカ深さ] フィールドに「-1」と表示されます。[Root] パーティションのレプリカが含まれている場合は、「0」と表示


されます。指定したサーバ上にレプリカがある場合は、[Root] に最も近いレプリカが [Root] からオブジェクト何個分離しているかを示す正の整数が表示されます。

eDirectory ツリー内のすべてのサーバは、同じタイムソースと同期している必要があります。そうしなければ、矛盾が発生したときに、レプリカ間でのオブジェクトの同期が正確に管理されなくなります。

同期修復ウィザードでは、各サーバのタイムソースはレポートできません。ただし、タイムサーバのタイプは報告します。この情報を参照すれば、時刻同期が正確に設定されているかどうかを確認できます。


**重要:** 時刻同期のステータス「Nearly-In-Sync(ほぼ同期)」を監視するには、DSRepair ではなく Novell iMonitor を使用してください。詳細については、7 章 171 ページの、「Novell iMonitor 2.1 の使用」を参照してください。

詳細については、85 ページの「ネットワーク時刻の同期」を参照してください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [同期修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [時刻同期] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## 即時同期をスケジュールリングする

この操作では、すべてのレプリカの同期を即座に行うようスケジュールリングします。この操作は、同期プロセスが通常のスケジュールどおりに実行されるのを待つことなく、同期情報を確認したい場合に使用します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory Maintenance Utilities(eDirectory 保守ユーティリティ)] > [同期修復] の順にクリックします。
- 3 操作を実行するサーバを指定し、[次へ] をクリックします。
- 4 操作を実行するサーバのユーザ名、パスワード、およびコンテキストを指定し、[次へ] をクリックします。
- 5 [即時同期のスケジュールリング] をクリックし、[次へ] をクリックします。
- 6 表示される指示に従って、操作を完了します。

## DSRepair のカスタムオプション

各 eDirectory プラットフォームの DSRepair ユーティリティには、Novell iManager で利用可能な修復機能のほかにも、通常の操作からは見えない拡張機能がいくつかあります。これらの拡張機能は、さまざまなプラットフォームで DSRepair ユーティリティをロードする際にスイッチを使用して利用できます。

- ◆ 233 ページの「eDirectory サーバ上で DSRepair を実行する」
- ◆ 234 ページの「DSRepair コマンドラインオプション」
- ◆ 235 ページの「DSRepair 詳細設定スイッチの使用」

# eDirectory サーバ上で DSRepair を実行する

## NetWare の場合

DSRepair を実行するには、サーバコンソールで「**srepair.nlm**」と入力します。

カスタムオプションで DSRepair を開くには、サーバコンソールで「**dsrepair -a**」と入力します。

## Windows の場合

1 [スタート] > [設定] > [コントロールパネル] > [Novell eDirectory Services] の順にクリックします。

2 dsrepair.dlm をクリックして、[開始] をクリックします。

カスタムオプションで DSRepair を開くには、Novell eDirectory Services コンソールの [起動パラメータ] フィールドに「**-a**」と入力し、dsrepair.dlm を開始します。

## Linux、Solaris、AIX、および HP-UX の場合

DSRepair を実行するには、サーバコンソールに次の構文を使用して「**ndsrepair**」と入力します。

```
ndsrepair { -U | -E | -C | -P [-Ad] | -S [-Ad] | -N | -T | -J entry_id | --version } [-F filename] [-A yes|no] [-O yes|no]
```

または

```
ndsrepair -R [-l yes|no] [-u yes|no] [-m yes|no] [-i yes|no] [-f yes|no] [-d yes|no] [-t yes|no] [-o yes|no] [-r yes|no] [-v yes|no] [-c yes|no] [-F filename] [-A yes|no] [-O yes|no]
```

**重要:** -Ad オプションは、Novell サポート担当者からの事前の指示がない限り使用しないでください。

### 例

無人修復を実行し、/root/ndsrepair.log ファイルにイベントを記録する場合 (またはログファイルがすでに存在していればそのログファイルに追加してイベントを記録する場合は)、次のコマンドを入力します。

```
ndsrepair -U -A no -F /root/ndsrepair.log
```

カスタムオプションで DSRepair を開くには、次のコマンドを入力します。

```
ndsrepair -Ad
```

すべてのグローバルスキーマ操作とその詳細設定オプションのリストを表示するには、次のコマンドを入力します。

```
ndsrepair -S -Ad
```

データベースを強制ロックしてローカルデータベースを修復するには、次のコマンドを入力します。

```
ndsrepair -R -l yes
```

**注:** ndsrepair コマンドの入力内容は、オプションファイルによってリダイレクトできます。オプションファイルは、レプリカおよびパーティション操作に関連するオプションやサブオプションを含むテキストファイルです。これらはサーバに対する認証を必要としません。各オプションまたはサブオプションは改行によって区切られます。ファイルの内容が、適切な順序で指定されていることを確認します。適切な順序になっていないと、予期しない結果が発生する場合があります。

## DSRepair コマンドラインオプション

オプション	説明
-U	[標準修復] オプションです。ユーザの操作なしに ndsrepair を実行または終了します。Novell サポート担当者から特定の操作を手動で実行するように指示された場合を除き、この修復方法の使用をお勧めします。修復が完了したらログファイルをチェックして、ndsrepair で変更された内容を確認します。
-P	[レプリカ操作とパーティション操作] オプションです。現在のサーバの eDirectory データベースファイルにレプリカが保存されているパーティションが表示されます。[レプリカオプション] メニューには、「レプリカの修復」、「パーティション操作のキャンセル」、「同期のスケジューリング」、および「ローカルレプリカをマスタレプリカとして指定」を実行するオプションがあります。
-S	[グローバルスキーマの操作] オプションです。このオプションには、このサーバのスキーマをツリーオブジェクトのマスタに準拠させるのに必要なスキーマ操作がいくつか含まれています。ただし、これらの操作は必要なときだけ使用してください。スキーマは、ローカル修復操作および無人修復操作によってすでに検査されています。
-C	[外部参照オブジェクトのチェック] オプションです。各外部参照オブジェクトをチェックして、そのオブジェクトを含むレプリカがあるかどうかを調べます。オブジェクトのあるパーティションレプリカを含むすべてのサーバがアクセス不能の場合、オブジェクトは見つかりません。オブジェクトが見つからない場合、警告が表示されます。
-E	[レプリカ同期のレポート] オプションです。現在のサーバ上にレプリカを持つすべてのパーティションのレプリカ同期ステータスをレポートします。この操作により、パーティションのレプリカを保持する各サーバ上のレプリカのツリーオブジェクトから同期ステータス属性が読み込まれます。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。12 時間以内に同期が完了していない場合は、警告メッセージが表示されます。
-N	[このデータベースに認識されているサーバ] オプションです。ローカル eDirectory データベースに認識されているすべてのサーバが表示されます。現在のサーバにツリーパーティションのレプリカがある場合、このサーバには eDirectory ツリー内のすべてのサーバのリストが表示されます。サーバオプションを実行するサーバを 1 つ選択します。
-J	ローカルサーバ上の 1 つのオブジェクトを修復します。修復するオブジェクトのエントリ ID(16 進形式) を指定する必要があります。破損している 1 つの特定のオブジェクトを修復するには、[標準修復 (-U)] オプションの代わりに、このオプションを使用できます。データベースのサイズによっては、[標準修復] オプションの完了に何時間もかかる場合があります。このオプションを使用して、時間を節約することができます。
-T	[時刻同期] オプションです。ローカル eDirectory データベースに登録されているすべてのサーバにアクセスして、各サーバの時刻同期ステータスの情報を要求します。このサーバにツリーパーティションのレプリカがある場合は、eDirectory ツリー内のすべてのサーバがポーリングされます。各サーバ上で実行されている eDirectory のバージョンもレポートされます。
-A	既存のログファイルに付加します。既存のログファイルに情報が追加されます。このオプションは、デフォルトで有効に設定されています。
-O	出力をファイルに記録します。このオプションは、デフォルトで有効に設定されています。

オプション	説明
-F <i>filename</i>	出力を指定したファイルに記録します。
-R	[ローカルデータベースの修復] オプションです。ローカル eDirectory データベースを修復します。eDirectory でオープンおよびアクセスできるように、修復操作を使用してローカルデータベースの矛盾を解決します。このオプションには、データベースの修復操作を容易にするサブオプションがあります。このオプションにはファンクション修飾子があります。ファンクション修飾子については、次の表で説明します。

-R オプションで使用するファンクション修飾子を次に示します。

オプション	説明
-l	修復操作中に eDirectory データベースをロックします。
-u	修復操作中に一時 eDirectory データベースを使用します。
-m	修復されていない元のデータベースを維持します。
-i	eDirectory データベース構造とインデックスをチェックします。
-f	データベースの空き領域を増やします。
-d	データベース全体を再構築します。
-t	ツリー構造のチェックを実行します。データベース内での接続状況が正しいかどうかを調べるため、ツリー構造のリンクをすべてチェックするには、「はい」を設定します。チェックを省略するには、「いいえ」を設定します。デフォルト値は「はい」です。
-o	オペレーショナルスキーマを再構築します。
-r	すべてのローカルレプリカを修復します。
-v	ストリームファイルを確認します。
-c	ローカル参照をチェックします。

## DSRepair 詳細設定スイッチの使用

**警告：**このセクションで説明する機能は、正しく使用しないと eDirectory ツリーが破損して元に戻せないことがあります。Novell サポート担当者からの指示がない限り、この機能は使用しないでください。

生産環境でこれらのうちのいずれかの機能を使用する前に、あらかじめサーバ上の eDirectory のフルバックアップをとっておくことをお勧めします。詳細については、[14 章 381 ページの、「Novell eDirectory のバックアップと復元」](#)を参照してください。

NetWare では、DSRepair(dsrepair -XK2 など) をロードするときに、サーバコンソールでこれらのオプションを使用します。

Linux、Solaris、AIX、および HP-UX では、「**ndsrepair -R -Ad -XK2**」と入力します。

Windows では、dsrepair.dlm を開始する前に、NDS コンソールの [起動パラメータ] フィールドにこれらのオプションを入力します。詳細については、[233 ページの「eDirectory サーバ上で DSRepair を実行する」](#)を参照してください。

スイッチ	説明
-NLC	NetWare サーバで「STORE NETWARE 5 CONN SCL MLA USAGE IN NDS」セットパラメータがオンになった場合、「NLS:CERT PEAK USED POOL」属性は非常に大きい値を取得する場合があります。DSRepair を -NLC オプションで実行すると、こうした大きい値はクリアされます。
-P	タイプが不明な eDirectory オブジェクトをすべて、参照済みとしてマークします。参照されたオブジェクトは、eDirectory レプリカ同期処理の対象にはなりません。
-WM	多くの場合、WM は次のようになります。ZENworks® 2.0 を使用すると「Registered Workstations(登録されたワークステーション)」属性は大きい値になります。DSRepair を -WM オプションで実行すると、こうした大きい値はクリアされます。
-XK2	このサーバの eDirectory データベース内にある eDirectory オブジェクトをすべて削除します。この操作では、破損したレプリカがどんな方法を使っても削除できない場合に、破損レプリカを削除できます。
-XK3	このサーバの eDirectory データベース内にある外部参照をすべて削除します。この操作では、機能していないレプリカ内の外部参照をすべて削除できます。参照が原因で問題が発生している場合、レプリカが再度機能するために、eDirectory は参照を再作成できます。

## eMBox クライアントを使用したデータベースの修復

eMBox(eDirectory Management Toolbox) クライアントはコマンドライン Java クライアントで、これを使用すると DSRepair にリモートアクセスできます。eMBox クライアントはバッチモードで実行できるため、これを使用して eDirectory DSRepair eMTool で無人修復を行うことができます。

eDirectory の一部としてサーバ上に `emboxclient.jar` ファイルがインストールされます。JVM を実行しているコンピュータならどこからでも実行できます。eMBox クライアントの詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

### DSRepair eMTool を使用する

- 1 非対話式モードで eMBox クライアントを実行するには、コマンドラインに次のコマンドを入力します。

```
java -cp path_to_the_file/emboxclient.jar embox -i
```

(クラスパスにすでに `emboxclient.jar` ファイルが含まれている場合、「`java embox -i`」と入力するだけです)。

次の eMBox クライアントのプロンプトが表示されます。

```
eMBox Client>
```

- 2 修復するサーバにログインするには、次のように入力します。

```
login -sserver_name_or_IP_address -pport_number  
-uusername.context -wpassword -n
```

ポート番号は通常 80 または 8008 です。ただし、すでにそのポートを使用している Web サーバが存在する場合は異なります。-n オプションを使用すると、セキュリティ保護されていない接続を開始します。

eMBox クライアントはログインが成功したかどうかを表示します。



- 3 次の構文を使用して修復コマンドを入力します。

```
dsrepair.task options
```

次に例を示します。

```
dsrepair.ufr は無人修復を実行します。
```

`dsrepair.rld -a -v` は、[すべてのローカルレプリカを修復] オプションおよび [ローカル参照をチェックする] オプションを使用して、ローカルデータベースを修復します。

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。

eMBox クライアントは修復が成功したかどうかを表示します。

DSRepair eMTool オプションの詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

- 4 eMBox クライアントからログアウトするには、次のコマンドを入力します。

```
logout
```

- 5 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

## DSRepair eMTool のオプション

次の表に DSRepair eMTool のオプションを示します。eMBox クライアントで `list -tdsrepair` コマンドを使用して DSRepair オプションの詳細を表示することもできます。詳細については、[483 ページの「eMTool とそのサービスを表示する」](#)を参照してください。

オプション	説明
rso	単一オブジェクトの修復
-o	オブジェクト ID(16 進数)
-d	オブジェクト DN
rts	時刻の同期
rss	すべてのパーティションの同期ステータスのレポート
rld	ローカルデータベースの修復
-l	修復中は常に eDirectory データベースをロック
-t	修復中に一時的な eDirectory データベースを使用
-d	修復されていない元のデータベースを維持
-p	データベース構造のチェックを実行
-i	データベースの構造チェックとインデックスチェックを実行
-f	データベースの未使用領域を増やす
-e	データベース全体を再構築
-c	ツリー構造のチェックを実行
-o	オペレーショナルスキーマを再構築
-a	すべてのローカルレプリカを修復
-m	メールディレクトリおよびストリームファイルを確認
-v	ローカル参照をチェック
ufr	無人修復

オプション	説明
rsn -o -d	選択したサーバのネットワークアドレスの修復 オブジェクト ID(16 進数) オブジェクト DN
ran	すべてのネットワークアドレスの修復
rsr -p -d	選択したレプリカの修復 パーティション ID パーティション DN
rer	すべてのレプリカの修復
ror -p -d	選択したレプリカリングの修復 パーティション ID パーティション DN
rar	すべてのレプリカのレプリカリングの修復
ssa -p -d	すべてのサーバのレプリカ同期ステータスのレポート パーティション ID パーティション DN
cer	外部参照のチェック
rao -p -d -s -d	このレプリカのすべてのオブジェクトを受信 パーティション ID パーティション DN サーバ ID サーバ DN
sao -p -d -s -d	リング内のすべてのレプリカにすべてのオブジェクトを送信 パーティション ID パーティション DN サーバ ID サーバ DN
dne -p -d	タイムスタンプの修復と新規エポックの宣言 パーティション ID パーティション DN
sri -p -d	即時同期のスケジューリング パーティション ID パーティション DN サーバ ID サーバ DN
sk -p -d -s -d	選択したサーバのレプリカを同期 パーティション ID パーティション DN サーバ ID サーバ DN
ske -p -d	すべてのサーバのレプリカを同期 パーティション ID パーティション DN

オプション	説明
dsr	選択したレプリカの削除
-p	パーティション ID
-d	パーティション DN
xsr	レプリカリングからサーバを削除
-p	パーティション ID
-d	パーティション DN
-s	サーバ ID
-d	サーバ DN
dnm	このサーバを新しいマスタレプリカに設定
-p	パーティション ID
-d	パーティション DN
dul	不明リーフオブジェクトの削除



# 10 WAN トラフィックマネージャ

WAN トラフィックマネージャ (WTM) を使用して WAN リンク上のレプリケーショントラフィックを管理することにより、ネットワークコストを削減できます。WAN トラフィックマネージャは Novell® eDirectory™ のインストール時にインストールされ、次の要素から構成されます。

- ◆ WTM

レプリカリング内の各サーバ上に常駐します。eDirectory がサーバ間トラフィックを送信する前に、WTM は WAN トラフィックポリシーを読み込んで、そのトラフィックを送信するかどうかを決定します。

- ◆ WAN トラフィックポリシー

eDirectory トラフィックの生成を制御する規則です。WAN トラフィックポリシーは、サーバオブジェクトまたは LAN エリアオブジェクト、あるいはその両方に eDirectory プロパティ値として保存されるテキストです。

- ◆ WANMAN Novell iManager プラグイン

この WTM とのインタフェースは、ポリシーの作成または変更、LAN エリアオブジェクトの作成、および LAN エリアまたはサーバへのポリシーの適用のために使用します。WTM を (eDirectory インストールの一部として) インストールする場合、スキーマには LAN エリアオブジェクト、およびサーバオブジェクト上の WAN トラフィックマネージャページが含まれます。

WAN トラフィックマネージャ (NetWare® 上では wtm.nlm、Windows 上では wtm.dlm) は、トラフィックを制御する各サーバ上に常駐する必要があります。パーティションのレプリカリングに 1 つの広域リンクの両側に配置されているサーバが含まれる場合は、そのレプリカリング内のすべてのサーバ上に WAN トラフィックマネージャをインストールします。

**重要:** Linux、Solaris、AIX、または HP-UX システムでは、WAN トラフィックマネージャはサポートされていません。

## WAN トラフィックマネージャについて

eDirectory などのネットワークディレクトリにより、サーバ間トラフィックが生成されます。広域ネットワーク (WAN) リンク上のトラフィックを管理しない場合、コストが不必要に増大したり、利用率の高い時間帯に通信速度が遅い WAN リンクに過大な負荷がかかたりすることがあります。

WAN トラフィックマネージャを使用すると、eDirectory により生成された (WAN リンク上の) サーバ間トラフィックや、eDirectory ツリー内の任意のサーバ間 eDirectory トラフィックを制御できます。WTM は、トラフィックのコスト、時刻、eDirectory 操作のタイプ、あるいはそのいずれかの組み合わせに基づいてトラフィックを制限できます。

たとえば、利用率の高い時間帯には WAN リンク上の eDirectory トラフィックを制限できます。この場合には、高帯域幅のアクティビティが利用率の低い時間帯に移行します。

レプリカの同期トラフィックを利用率の低い時間帯だけに制限することにより、コストを削減することもできます。

WAN トラフィックマネージャは、eDirectory により開始された定期的なイベント (レプリカの同期など) だけを制御します。管理者またはユーザにより開始されたイベントや、非 eDirectory サーバ間トラフィック (時刻の同期など) は制御しません。

サーバ間トラフィックを生成する eDirectory プロセスを次の表に示します。

プロセス	説明
レプリカの同期	<p>eDirectory オブジェクトへの変更がパーティションのすべてのレプリカで同期されるようにします。指定されたパーティションのコピーを保持するすべてのサーバは、変更を同期するために他のサーバと通信する必要があります。</p> <p>実行されるレプリカの同期には 2 つのタイプがあります。</p> <ul style="list-style-type: none"><li>◆ 即時同期は、eDirectory オブジェクトが変更されるか、あるいはディレクトリツリー内でオブジェクトが追加または削除されると実行されます。</li><li>◆ 遅延同期は、複数の eDirectory オブジェクトに共通した変更で、繰り返し実行される特定の変更 (ログインプロパティへの変更など) の後で実行されます。このような変更の例としては、ユーザがログインまたはログアウトするときのログイン時刻、最終ログイン時刻、ネットワークアドレス、およびリビジョンの各プロパティへの更新があります。</li></ul> <p>遅延同期プロセスは、即時同期プロセスが実行されない場合にのみ実行されます。デフォルトでは、即時同期は変更が保存されてから 10 秒後に実行され、遅延同期は他の変更が発生してから 22 分後に実行されます。</p>
スキーマの同期	<p>ディレクトリツリー内の各パーティションでスキーマが整合性を保ち、すべてのスキーマ変更がネットワーク上で更新されるようにします。</p> <p>デフォルトでは、このプロセスは 4 時間に 1 回実行されます。</p>
ハートビート	<p>ディレクトリオブジェクトがパーティションのすべてのレプリカで整合性を保つようにします。パーティションのコピーを持つすべてのサーバは、整合性をチェックするために、パーティションを持つ他のサーバと通信する必要があります。</p> <p>デフォルトでは、このプロセスはパーティションのレプリカを含むすべてのサーバ上で 30 分に 1 回実行されます。</p>
リンバ	<p>サーバ名またはサーバアドレスが変更されたときに、そのサーバのレプリカポインタテーブルが更新されるようにします。この変更は次のような場合に発生します。</p> <ul style="list-style-type: none"><li>◆ autoexec.ncf ファイルで新しいサーバ名または IPX™ 内部アドレスを使用してサーバをリブートした。</li><li>◆ プロトコルを追加するためにアドレスを追加した。</li></ul> <p>サーバがブートされると、リンバプロセスにより、サーバ名およびサーバの IPX アドレスがレプリカポインタテーブルに保存されているデータと比較されます。サーバ名とアドレスのどちらかまたは両方が一致しない場合、eDirectory は、そのサーバの項目を含むすべてのレプリカポインタテーブルを自動的に更新します。</p> <p>リンバプロセスでは、レプリカリング内の各サーバのツリー名が正しいかどうかもチェックされます。</p> <p>リンバプロセスはサーバがブートされてから 5 分後に実行され、その後 3 時間に 1 回実行されます。</p>


プロセス	説明
バックリンク	サーバ上のレプリカに保存されていない eDirectory オブジェクトへのポインタである外部参照を確認します。通常の場合、バックリンクプロセスはローカルデータベースがオープンされてから 2 時間後に実行され、その後 13 時間に 1 回実行されます。
接続管理	レプリカリング内のサーバが NCP™ パケットを転送するには、セキュリティレベルの高い接続が必要です。接続管理プロセスにより、仮想クライアント接続と呼ばれる安全な接続が確立されます。  場合によっては、接続管理プロセスにより、スキーマの同期プロセスまたはバックリンクプロセスのための仮想クライアント接続も確立する必要があります。時刻サービスの環境設定によっては、時刻の同期プロセスでも仮想クライアント接続が必要になります。
サーバステータスチェック	サーバステータスチェックは、レプリカを持たない各サーバ上で開始されます。サーバオブジェクトを含むパーティションの書き込み可能なレプリカを保持するサーバのうち、最も近くにあるサーバへの接続が確立されます。  サーバステータスチェックは 6 分に 1 回実行されます。

## LAN エリアオブジェクト

LAN エリアオブジェクトを使用すると、サーバグループの WAN トラフィックポリシーを簡単に管理できます。LAN エリアオブジェクトを作成した後で、LAN エリアオブジェクトにサーバを追加したり、LAN エリアオブジェクトからサーバを削除することができます。LAN エリアに適用されたポリシーは、その LAN エリア内のすべてのサーバに適用されます。

広域リンクによって他の LAN に接続している LAN 内に複数のサーバがある場合は、LAN エリアオブジェクトを作成します。LAN エリアオブジェクトを作成しない場合は、各サーバの WAN トラフィックを個別に管理する必要があります。

### LAN エリアオブジェクトを作成する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [LAN エリアの作成] の順にクリックします。
- 3 [オブジェクトクラス] ドロップダウンリストの [WANMAN-LAN エリア] を選択します。
- 4 オブジェクトの名前とコンテキストを指定して、[作成] をクリックします。


次のいずれかのセクションに進みます。

- ◆ [243 ページの「LAN エリアオブジェクトへサーバを追加する」](#)
- ◆ [245 ページの「WAN ポリシーを適用する」](#)

### LAN エリアオブジェクトへサーバを追加する

1 つのサーバは 1 つの LAN エリアオブジェクトにのみ属することができます。追加するサーバが LAN エリアオブジェクトにすでに属している場合、そのサーバはそのオブジェクトから削除され、新しいオブジェクトに追加されます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] の順にクリックします。

- 3 [LAN エリアの表示] をクリックし、サーバを追加する LAN エリアオブジェクトをクリックします。
- 4 [サーバリスト] をクリックし、[ オブジェクト セレクタ ] ボタン  をクリックします。
- 5 目的のサーバを選択します。
- 6 追加する各サーバについて、[ステップ 4](#) から [ステップ 5](#) を繰り返します。  
WAN ポリシーを LAN エリアオブジェクトに適用して、そのポリシーをグループ内のすべてのサーバに適用する方法の詳細については、[245 ページ](#)の「[WAN ポリシーを適用する](#)」を参照してください。
- 7 [適用] をクリックし、[OK] をクリックします。

## LAN エリアオブジェクトへ追加情報を追加する

ConsoleOne<sup>®</sup> を使用して、LAN エリアオブジェクトに記述情報を追加できます。この機能は、Novell iMonitor では使用できません。

- 1 ConsoleOne で、LAN エリアオブジェクトを右クリックします。
- 2 [プロパティ] > [一般] の順にクリックします。
- 3 必要に応じて、所有者、説明、地域、部署、および組織の情報を追加します。
- 4 [適用] をクリックし、[OK] をクリックします。

## WAN トラフィックポリシー

WAN トラフィックポリシーは、eDirectory トラフィックの生成を制御する一連の規則です。この規則はテキストとして作成され、サーバオブジェクトまたは LAN エリアオブジェクト、あるいはその両方に eDirectory プロパティ値として保存されます。ポリシーは、単純な処理言語に基づいて解釈されます。

ポリシーは個々のサーバに適用できます。LAN エリアオブジェクトを作成して、このオブジェクトに複数のサーバを割り当てることもできます。LAN エリアオブジェクトに適用されたポリシーは、そのオブジェクトに割り当てられたすべてのサーバに自動的に適用されます。

WAN トラフィックマネージャには、いくつかの定義済みポリシーグループがあります。これらのポリシーは、そのまま使用することも、必要に応じて変更することもでき、あるいは新しいポリシーを作成することもできます。

- ◆ [245 ページ](#)の「[WAN ポリシーを適用する](#)」
- ◆ [246 ページ](#)の「[WAN ポリシーを変更する](#)」
- ◆ [247 ページ](#)の「[既存のポリシーをリネームする](#)」
- ◆ [247 ページ](#)の「[新しい WAN ポリシーを作成する](#)」

## 定義済みポリシーグループ

類似した機能を持つ定義済みポリシーのグループを次の表に示します。

ポリシーグループ	説明
1-3AM.WMG	トラフィックの送信が午前 1 時から午前 3 時までの時間帯だけに制限されます。




ポリシーグループ	説明
7AM-6PM.WMG	トラフィックの送信が午前7時から午後6時までの時間帯だけに制限されます。
COSTLT20.WMG	コストファクタが 20 より小さいトラフィックのみ送信が許可されます。
IPX.WMG	IPX トラフィックのみ許可されます。
NDSTTYP.S.WMG	さまざまなeDirectoryトラフィックタイプのサンプルポリシーが提供されます。
OPNSPOOF.WMG	既存の WAN 接続のみ使用が許可されます。
OPNSPOOF.WMG	既存の WAN 接続のみ使用が許可されますが、15 分間使用されていない接続は無効と見なされます。この場合は使用できません。
SAMEAREA.WMG	同じネットワークエリア内のトラフィックのみ許可されます。
TCPIP.WMG	TCP/IP トラフィックのみ許可されます。
TIMECOST.WMG	すべてのトラフィックが午前 1 時から午前 1 時 30 分までの時間帯だけに制限されますが、同じロケーション内のサーバには連続的な対話が許可されます。

定義済みポリシーグループおよび個々のポリシーの詳細については、[250 ページの「WAN トラフィックマネージャポリシーグループ」](#)を参照してください。

## WAN ポリシーを適用する

WAN ポリシーを個々のサーバまたは LAN エリアオブジェクトに適用できます。個々のサーバに適用されたポリシーは、そのサーバの eDirectory トラフィックのみ管理します。LAN エリアオブジェクトに適用されたポリシーは、そのオブジェクトに属しているすべてのサーバのトラフィックを管理します。

WAN トラフィックマネージャは、WANMAN.INI の WAN ポリシーグループセクションに格納されている「key = values」ステートメントを参照します。「key」はスナップインに表示されるポリシー名で、「value」は区切られたポリシーを格納するテキストファイルへのパスです。


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] の順にクリックします。
- 3 [LAN エリアの表示] をクリックし、LAN エリアオブジェクトをクリックします。  
または  
[NCP サーバの表示] をクリックし、NCP サーバオブジェクトをクリックします。
- 4 [ポリシーの追加] をクリックし、使用するポリシーグループを選択します。  
詳細については、[244 ページの「定義済みポリシーグループ」](#)を参照してください。
- 5 [OK] をクリックします。  
ポリシーグループからロードされたポリシーのリストが表示されます。
- 6 [OK] をクリックします。  
ポリシーの内容を参照したり、ポリシーを変更することができます。ポリシー内のエラーをチェックするには、[ポリシーのチェック] をクリックします。

- 7 不要なポリシーを削除するには、[ポリシー名] ドロップダウンリストでポリシーを選択し、[ポリシーの削除] をクリックします。
- 8 [適用] をクリックし、[OK] をクリックします。


## WAN ポリシーを変更する

WAN トラフィックマネージャが提供する定義済みポリシーグループは、要件に合わせて変更できます。また、独自に作成したポリシーを変更することもできます。

### サーバに適用された WAN ポリシーを変更する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] > [NCP サーバの表示] の順にクリックします。
- 3 編集するポリシーを含むサーバオブジェクトをクリックします。
- 4 [ポリシー名] ドロップダウンリストで、編集するポリシーを選択します。
- 5 [ポリシー] フィールドで必要に応じてポリシーを編集します。  
WAN ポリシーの構成の詳細については、[265 ページの「WAN ポリシーの構成」](#)を参照してください。  
WAN ポリシーの構文の詳細については、[268 ページの「ポリシーセクションで使用される構文」](#)を参照してください。
- 6 [ポリシーのチェック] をクリックすると、構文または構成のエラーが識別されます。  
WAN トラフィックマネージャは、エラーを含むポリシーを実行しません。
- 7 変更した場合は、[適用] をクリックします。
- 8 不要なポリシーを削除するには、[ポリシー名] ドロップダウンリストでポリシーを選択し、[ポリシーの削除] をクリックします。
- 9 [適用] をクリックし、[OK] をクリックします。

### LAN エリアオブジェクトに適用された WAN ポリシーを変更する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] > [LAN エリアの表示] の順にクリックします。
- 3 編集するポリシーを含む LAN エリアオブジェクトをクリックします。
- 4 [ポリシー名] ドロップダウンリストで、編集するポリシーを選択します。
- 5 [ポリシー] フィールドで必要に応じてポリシーを編集します。  
WAN ポリシーの構成の詳細については、[265 ページの「WAN ポリシーの構成」](#)を参照してください。  
WAN ポリシーの構文の詳細については、[268 ページの「ポリシーセクションで使用される構文」](#)を参照してください。
- 6 [ポリシーのチェック] をクリックすると、構文または構成のエラーが識別されます。  
WAN トラフィックマネージャは、エラーを含むポリシーを実行しません。
- 7 変更した場合は、[適用] をクリックします。

- 8 不要なポリシーを削除するには、[ポリシー名] ドロップダウンリストでポリシーを選択し、[ポリシーの削除] をクリックします。
- 9 [適用] をクリックし、[OK] をクリックします。


### 既存のポリシーをリネームする

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] の順にクリックします。
- 3 [LAN エリアの表示] をクリックし、LAN エリアオブジェクトをクリックします。  
または  
[NCP サーバの表示] をクリックし、NCP サーバオブジェクトをクリックします。
- 4 [ポリシー名] ドロップダウンリストで、リネームするポリシーを選択します。
- 5 [ポリシーの名前の変更] をクリックし、新しい名前を指定します。  
名前は完全識別名である必要があります。
- 6 [OK] をクリックし、[適用]、[OK] の順にクリックします。


### 新しい WAN ポリシーを作成する

サーバオブジェクトまたは LAN エリアオブジェクトに適用する WAN ポリシーを作成できます。個々のサーバ用に作成されたポリシーは、そのサーバの eDirectory トラフィックのみ管理します。LAN エリアオブジェクト用に作成されたポリシーは、そのオブジェクトに属しているすべてのサーバのトラフィックを管理します。

### サーバオブジェクトの WAN ポリシーを作成する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] > [NCP サーバの表示] の順にクリックします。
- 3 新しいポリシーを作成するサーバオブジェクトをクリックし、[ポリシーの作成] をクリックします。
- 4 新しいポリシーの名前を指定し、[OK] をクリックします。  
指定する名前は完全識別名である必要があります。
- 5 [ポリシー] テキストボックスに必要な情報を指定します。  
WAN ポリシーの構成の詳細については、[265 ページの「WAN ポリシーの構成」](#)を参照してください。  
WAN ポリシーの構文の詳細については、[268 ページの「ポリシーセクションで使用する構文」](#)を参照してください。
- 6 [適用] をクリックし、[OK] をクリックします。

### LAN エリアオブジェクトの WAN ポリシーを作成する


- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] > [LAN エリアの表示] の順にクリックします。

- 3 WANポリシーを作成するLANエリアオブジェクトをクリックし、[ポリシーの作成] をクリックします。
- 4 新しいポリシーの名前を指定し、[OK] をクリックします。  
指定する名前は完全識別名である必要があります。
- 5 [ポリシー] テキストボックスに必要な情報を指定します。  
WAN ポリシーの構成の詳細については、265 ページの「WAN ポリシーの構成」を参照してください。  
WAN ポリシーの構文の詳細については、268 ページの「ポリシーセクションで使用される構文」を参照してください。
- 6 [適用] をクリックし、[OK] をクリックします。

## WAN トラフィックを制限する

WAN トラフィックマネージャには、トラフィックを特定の時間帯だけに制限する 2 つの定義済みポリシーグループがあります ( 詳細については、250 ページの「1-3AM.WMG」および 251 ページの「7AM-6PM.WMG」を参照してください)。これらのポリシーを変更すると、任意の時間帯を選択して、トラフィックをその時間帯だけに制限できます。

ここでは午前 1 時から午前 3 時までのグループを変更する場合について説明しますが、午前 7 時から午後 6 時までのグループを変更する場合でも同様の手順を使用できます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック] > [WAN トラフィックマネージャの概要] の順にクリックします。
- 3 [LAN エリアの表示] をクリックし、LAN エリアオブジェクトをクリックします。  
または  
[NCP サーバの表示] をクリックし、NCP サーバオブジェクトをクリックします。
- 4 [ポリシーの追加] をクリックします。
- 5 定義済みポリシーのリストの中から 1-3AM.WMG を選択し、[OK] を 2 回クリックします。

ポリシーが [ポリシー] テキストボックスに表示されるので、ここでポリシーを変更します。たとえば、トラフィックを許可する時間帯を午前 1 時から午前 3 時までではなく午前 2 時から午後 5 時までに設定するには、次のように変更します。

```
/* This policy limits all traffic to between 2 and 5 pm */
LOCAL BOOLEAN Selected;
SELECTOR
    Selected := Now.hour >= 2 AND Now.hour < 17;
    IF Selected THEN
        RETURN 50; /* between 2am and 5pm this policy has a
high priority */
    ELSE
        RETURN 1; /* return 1 instead of 0 in case there are
no other policies */
        /* if no policies return > 0, WanMan assumes
SEND */
    END
END
PROVIDER
    IF Selected THEN
```

```
        RETURN SEND; /* between 2am and 5pm, SEND */
    ELSE
        RETURN DONT_SEND; /* other times, don't */
    END
END
```

コメント行 (`/*` と `*/` で囲まれた部分) では、午前と午後を使用して時刻を指定できます。ただし、実行コードでは 24 時間形式を使用して指定する必要があります。したがって、午後 5 時は 17 になります。

WAN ポリシーの構成の詳細については、[265 ページの「WAN ポリシーの構成」](#)を参照してください。

WAN ポリシーの構文の詳細については、[268 ページの「ポリシーセクションで使用される構文」](#)を参照してください。

- 6 構文または構成のエラーを識別するには、ポリシーの構文を変更した後で[ポリシーのチェック] をクリックします。

ポリシーのチェックの結果が表示されます。

WAN トラフィックマネージャは、エラーを含むポリシーを実行しません。

- 7 元の 1-3 am ポリシーを変更しない場合は、新しいポリシーを別の名前を追加します。

**7a** [ポリシーの名前の変更] をクリックします。

**7b** 編集するポリシーの名前を指定し、[OK] をクリックします。

- 8 [適用] をクリックし、[OK] をクリックします。

## コストファクタを割り当てる

コストファクタにより、WAN トラフィックマネージャは特定の送信先とのトラフィックのコストを比較して、WAN ポリシーを使用してトラフィックを管理できます。WAN ポリシーでは、コストファクタを使用して WAN トラフィックの相対コストが決定されます。トラフィックを送信するか決定するのに、この情報を使用できます。

コストファクタは、時間単位あたりの費用として表されます。各 WAN トラフィックポリシーで一貫して同じ単位が使用されている限り、あらゆる単位でコストファクタを表すことができます。つまり、同じ比率のみを使用している限り、ドル / 時間、セント / 分、円 / 秒など、任意のコスト対時間の比率を使用できます。


トラフィックの相対コストを表す送信先コストファクタを、特定のアドレス範囲に割り当てることができます。したがって、サーバのグループ全体のコストを 1 つの宣言で割り当てることができます。デフォルトコストファクタを割り当てることもできます。デフォルトコストファクタは、送信先のコストが指定されていない場合に使用されます。

送信先に対してコストが割り当てられていない場合は、デフォルトコストが使用されます。サーバまたは LAN エリアオブジェクトに対してデフォルトコストが指定されていない場合は、-1 の値が割り当てられます。



コストファクタに基づいてトラフィックを制限するサンプルポリシーの詳細については、[251 ページの「COSTLT20.WMG」](#)を参照してください。

ポリシーを変更する方法の詳細については、[246 ページの「WAN ポリシーを変更する」](#)を参照してください。

## デフォルトコストファクタを割り当てる

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック管理] > [WAN トラフィックマネージャの概要] の順にクリックします。
- 3 [LAN エリアの表示] をクリックし、LAN エリアオブジェクトをクリックします。  
または  
[NCP サーバの表示] をクリックし、NCP サーバオブジェクトをクリックします。
- 4 [コスト] をクリックし、[デフォルトコスト] フィールドにコストを入力します。  
コストは負以外の整数でなければなりません。指定したデフォルトコストは、サーバまたは LAN エリアオブジェクトの送信先のうち、割り当てたコストの送信先アドレス範囲内に収まらないすべての送信先に割り当てられます。コストは、ドルなどの通貨単位で指定するか、または 1 秒あたりのパケット数で指定できます。
- 5 [適用] をクリックし、[OK] をクリックします。

## 送信先アドレス範囲へコストを割り当てる

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [WAN トラフィック管理] > [WAN トラフィックマネージャの概要] の順にクリックします。
- 3 [LAN エリアの表示] をクリックし、LAN エリアオブジェクトをクリックします。  
または  
[NCP サーバの表示] をクリックし、NCP サーバオブジェクトをクリックします。
- 4 [コスト] をクリックします。
- 5 [追加] ボタン  をクリックします。
- 6 WANMAN コストの作成ウィンドウで、[TCP/IP アドレスタイプ] または [IPX アドレスタイプ] を選択します。
- 7 範囲の開始アドレスと終了アドレスを TCP/IP または IPX の適切な形式で指定します。
- 8 [コスト] テキストフィールドで、コストを負以外の整数で指定します。
- 9 [OK] をクリックし、[適用]、[OK] の順にクリックします。

## WAN トラフィックマネージャポリシーグループ

WAN トラフィックマネージャには、次の定義済みポリシーグループがあります。

ポリシーグループを適用する方法の詳細については、[245 ページの「WAN ポリシーを適用する」](#)を参照してください。

### 1-3AM.WMG

このグループのポリシーでは、トラフィックを送信できる時間帯が午前 1 時から午前 3 時までだけに制限されます。次の 2 つのポリシーがあります。

- ◆ 1 - 3 am, NA

バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期がこの時間帯だけに制限されます。

- ◆ 1 - 3 am

他のすべてのトラフィックがこの時間帯だけに制限されます。

すべてのトラフィックをこの時間帯だけに制限するには、両方のポリシーを適用する必要があります。

## 7AM-6PM.WMG

このグループのポリシーでは、トラフィックを送信できる時間帯が午前 7 時から午後 6 時までだけに制限されます。次の 2 つのポリシーがあります。

- ◆ 7 am - 6 pm, NA

バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期がこの時間帯だけに制限されます。

- ◆ 7 am - 6 pm

他のすべてのトラフィックがこの時間帯だけに制限されます。

すべてのトラフィックをこの時間帯だけに制限するには、両方のポリシーを適用する必要があります。

## COSTLT20.WMG

このグループのポリシーでは、コストファクタが 20 未満のトラフィックのみ送信が許可されます。次の 2 つのポリシーがあります。

- ◆ Cost < 20, NA

コストファクタが 20 以上の場合、バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が抑止されます。

- ◆ Cost < 20

コストファクタが 20 以上の場合、他のすべてのトラフィックが抑止されます。

コストファクタが 20 以上のトラフィックをすべて抑止するには、両方のポリシーを適用する必要があります。

## IPX.WMG

このグループのポリシーでは、IPX トラフィックのみ送信が許可されます。次の 2 つのポリシーがあります。

- ◆ IPX, NA

IPX によって生成されたトラフィック以外の場合、バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が抑止されます。

- ◆ IPX

IPX によって生成されたトラフィック以外の場合、他のすべてのトラフィックが抑止されます。

IPX 以外のトラフィックをすべて抑止するには、両方のポリシーを適用する必要があります。

## NDSTTYP.S.WMG

このグループのポリシーは、さまざまな eDirectory トラフィックタイプのサンプルポリシーです。eDirectory がこのタイプの要求に渡す変数が格納されています。

- ◆ 252 ページの「Catch All with Addresses のサンプル」
- ◆ 252 ページの「Catch All without Addresses のサンプル」
- ◆ 252 ページの「NDS\_BACKLINK\_OPEN のサンプル」
- ◆ 253 ページの「NDS\_BACKLINKS のサンプル」
- ◆ 254 ページの「NDS\_CHECK\_LOGIN\_RESTRICTION のサンプル」
- ◆ 256 ページの「NDS\_CHECK\_LOGIN\_RESTRICTION\_OPEN のサンプル」
- ◆ 256 ページの「NDS\_JANITOR のサンプル」
- ◆ 258 ページの「NDS\_JANITOR\_OPEN のサンプル」
- ◆ 259 ページの「NDS\_LIMBER のサンプル」
- ◆ 260 ページの「NDS\_LIMBER\_OPEN のサンプル」
- ◆ 261 ページの「NDS\_SCHEMA\_SYNC のサンプル」
- ◆ 262 ページの「NDS\_SCHEMA\_SYNC\_OPEN のサンプル」
- ◆ 263 ページの「NDS\_SYNC のサンプル」

### Catch All with Addresses のサンプル

アドレスのあるトラフィックタイプのためのサンプルポリシーです。

### Catch All without Addresses のサンプル

アドレスのないトラフィックタイプのためのサンプルポリシーです。

### NDS\_BACKLINK\_OPEN のサンプル

NDS\_BACKLINK\_OPEN は、対応する NDS\_BACKLINKS クエリの実行中に CheckEachNewOpenConnection または CheckEachAlreadyOpenConnection が 1 に設定されていた場合にのみ使用されるトラフィックタイプです。

このクエリは、CheckEachNewOpenConnection が 1 で eDirectory がバックリンク処理のために新しい接続を確立する必要がある場合、または CheckEachAlreadyOpenConnection が 1 で eDirectory が既存の接続を再使用する必要がある場合には、常に生成されます。

- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 入力および出力、タイプ INTEGER)

ConnectionIsAlreadyOpen が TRUE の場合、ExpirationInterval は、既存の接続ですでに設定されている有効間隔に設定されます。ConnectionIsAlreadyOpen が TRUE 以外の場合、NDS\_BACKLINKS クエリで割り当てられた ExpirationInterval に設定されます。0 の値は、デフォルト (2 時間) を使用する必要があることを示します。終了時には、この変数の値が接続の有効間隔として割り当てられます。

値	説明
<0, 0	デフォルト有効間隔を使用 (デフォルト)。
>0	この接続に割り当てる有効間隔。



- ◆ **ConnectionIsAlreadyOpen**( 入力のみ、タイプ BOOLEAN)

eDirectory が既存の接続を再使用できる場合は、この変数は TRUE です。新しい接続を確立する必要がある場合は、この変数は FALSE です。

値	説明
TRUE	eDirectory は、このアドレスへの接続がすでに確立されていて、再使用できると判別します。
FALSE	eDirectory は、このアドレスへの接続を持たないため、新しい接続を確立する必要があります。

- ◆ **ConnectionLastUsed**( 入力のみ、タイプ TIME)

ConnectionIsAlreadyOpen が TRUE の場合、ConnectionLastUsed は、この接続を使用して eDirectory から最後にパケットが送信された時点です。ConnectionIsAlreadyOpen が TRUE 以外の場合、この値は 0 になります。

値	説明
TRUE	ConnectionLastUsed は、この接続上で eDirectory が最後にパケットを送信した時点です。
FALSE	ConnectionLastUsed は 0 になります。

## NDS\_BACKLINKS のサンプル

eDirectory は、バックリンクまたは外部参照をチェックする前に、このアクティビティが許容される時間帯かどうかを WAN トラフィックマネージャに照会します。NDS\_BACKLINKS には送信先アドレスがないため、NO\_ADDRESSES ポリシーが必要になります。WAN トラフィックマネージャが DONT\_SEND を返した場合は、バックリンクチェックは延期され、再度スケジューリングされます。次の変数が使用できます。

- ◆ **Last**( 入力のみ、タイプ TIME)

eDirectory 起動後にバックリンクチェックが最後に実行された時刻。eDirectory が起動されると、Last は 0 に初期化されます。NDS\_BACKLINKS が SEND を返した場合は、eDirectory によるバックリンク処理が完了した後で、Last は現在の時刻に設定されます。

- ◆ **Version**( 入力のみ、タイプ INTEGER)

eDirectory のバージョン。

- ◆ **ExpirationInterval**( 出力のみ、タイプ INTEGER)

バックリンク処理中に確立したすべての接続の有効間隔。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

◆ Next(出力のみ、タイプ TIME)

eDirectory が次のバックリンクチェック実行をスケジュールリングする時点を示します。

値	説明
過去、0	デフォルトスケジュールリングを使用。
将来	バックリンクをスケジュールリングする時点。

◆ CheckEachNewOpenConnection(出力のみ、タイプ INTEGER)

バックリンクの実行中に eDirectory が新しい接続を確立する必要が生じた場合の動作を、eDirectory に指示します。

CheckEachNewOpenConnection は 0 に初期化されます。

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します (デフォルト)。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

◆ CheckEachAlreadyOpenConnection(出力のみ、タイプ INTEGER)

この変数は、バックリンクの実行中に eDirectory がすでに確立されている接続を再使用する必要が生じた場合の動作を、eDirectory に指示します。

CheckEachAlreadyOpenConnection は 0 に初期化されます。

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します (デフォルト)。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

## NDS\_CHECK\_LOGIN\_RESTRICTION のサンプル

eDirectory は、ログイン制限をチェックする前に、このアクティビティが許容される時間帯かどうかを WAN トラフィックマネージャに照会します。トラフィックタイプ NDS\_CHECK\_LOGIN\_RESTRICTIONS には送信先アドレスがないため、NO\_ADDRESSES ポリシーが必要になります。WAN トラフィックマネージャが DONT\_SEND を返した場合は、エラーが発生して、チェックは実行されません。

次の変数が使用できます。

- ◆ Version( 入力のみ、タイプ INTEGER)

eDirectory のバージョン。

- ◆ Result( 出力のみ、タイプ INTEGER)

NDS\_CHECK\_LOGIN\_RESTRICTIONS の結果が DONT\_SEND である場合、次の値がオペレーティングシステムに返されます。

値	説明
0	ログインが許可されます。
1	現在の時間ブロック中は、ログインが許可されません。
2	アカウントが無効であるか、または有効期限が切れています。
3	アカウントが削除されています。

- ◆ ExpirationInterval( 出力のみ、タイプ INTEGER)

この接続に割り当てる有効間隔です。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

- ◆ CheckEachNewOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

- ◆ CheckEachAlreadyOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

## NDS\_CHECK\_LOGIN\_RESTRICTION\_OPEN のサンプル

NDS\_CHECK\_LOGIN\_RESTRICTION\_OPEN は、対応する NDS\_CHECK\_LOGIN\_RESTRICTIONS クエリの実行中に CheckEachNewOpenConnection または CheckEachAlreadyOpenConnection が 1 に設定されていた場合にのみ使用されます。このクエリは、CheckEachNewOpenConnection が 1 に設定されていて eDirectory が次の操作を必要とする場合には、常に生成されます。

- ◆ リンバを実行する前に新しい接続を確立する。
- ◆ ログイン制限をチェックする前に新しい接続を確立する。
- ◆ 既存の接続を再使用する。

次の変数が使用できます。

- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 入力および出力、タイプ INTEGER)

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

- ◆ ConnectionIsAlreadyOpen( 入力のみ、タイプ BOOLEAN)

値	説明
TRUE	eDirectory は、このアドレスへの接続がすでに確立されていて、再使用できると判別します。
FALSE	eDirectory は、このアドレスへの接続を持たないため、新しい接続を確立する必要があります。

- ◆ ConnectionLastUsed( 入力のみ、タイプ TIME)

ConnectionIsAlreadyOpen が TRUE の場合、ConnectionLastUsed は、この接続を使用して eDirectory から最後にパケットが送信された時点です。ConnectionIsAlreadyOpen が TRUE 以外の場合、この値は 0 になります。

値	説明
TRUE	ConnectionLastUsed は、この接続上で eDirectory が最後にパケットを送信した時点です。
FALSE	ConnectionLastUsed は 0 になります。

## NDS\_JANITOR のサンプル

eDirectory は、ジャンタを実行する前に、このアクティビティが許容される時間帯かどうかを WAN トラフィックマネージャに照会します。NDS\_JANITOR には送信先アドレスがないため、NO\_ADDRESSES ポリシーが必要になります。WAN トラフィックマネージャが DONT\_SEND を返した場合は、ジャンタ作業は延期され、再度スケジューリングされます。

次の変数が使用できます。

◆ Last( 入力のみ、タイプ TIME)

eDirectory 起動後にジャニタ作業が最後に実行された時刻。eDirectory が起動されると、Last は 0 に初期化されます。NDS\_JANITOR が「SEND」を返した場合は、eDirectory によるジャニタ処理が完了した後で、Last は現在の時刻に設定されます。

◆ Version( 入力のみ、タイプ INTEGER)

eDirectory のバージョン。

◆ ExpirationInterval( 出力のみ、タイプ INTEGER)

ジャニタの実行中に確立したすべての接続の有効間隔。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

◆ Next( 出力のみ、タイプ TIME)

ジャニタ作業を次回スケジューリングする時点を eDirectory に示します。

値	説明
過去、0	デフォルトスケジューリングを使用。
将来	ジャニタ作業をスケジューリングする時点。

◆ CheckEachNewOpenConnection( 出力のみ、タイプ INTEGER)

ジャニタの実行中に eDirectory が新しい接続を確立する必要がある場合の動作を、eDirectory に指示します。

CheckEachNewOpenConnection は 0 に初期化されます。

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

◆ CheckEachAlreadyOpenConnection( 出力のみ、タイプ INTEGER)

ジャニタの実行中に eDirectory がすでに確立されている接続を再使用する必要がある場合の動作を、eDirectory に指示します。

CheckEachAlreadyOpenConnection は 0 に初期化されます。

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します (デフォルト)。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

## NDS\_JANITOR\_OPEN のサンプル

NDS\_JANITOR\_OPEN は、対応する NDS\_JANITOR クエリの実行中に CheckEachNewOpenConnection または CheckEachAlreadyOpenConnection が 1 に設定されていた場合にのみ使用されます。このクエリは、CheckEachNewOpenConnection が 1 で eDirectory がバックリンク処理の前に新しい接続を確立する必要がある場合、または CheckEachAlreadyOpenConnection が 1 で eDirectory が既存の接続を再使用する必要がある場合には、常に生成されます。

次の変数が使用できます。

- ◆ Version( 入力のみ、タイプ INTEGER)

eDirectory のバージョン。

- ◆ ExpirationInterval( 入力および出力、タイプ INTEGER)

ConnectionIsAlreadyOpen が TRUE の場合、ExpirationInterval は、既存の接続ですでに設定されている有効間隔に設定されます。ConnectionIsAlreadyOpen が TRUE 以外の場合、NDS\_JANITOR クエリで割り当てられた ExpirationInterval に設定されます。0 の値は、デフォルト (2 時間、10 秒) を使用する必要があることを示します。終了時には、この変数の値が接続の有効間隔として割り当てられます。

値	説明
<0, 0	デフォルト有効間隔を使用 (デフォルト)。
>0	この接続に割り当てる有効間隔。

- ◆ ConnectionIsAlreadyOpen( 入力のみ、タイプ BOOLEAN)

eDirectory が既存の接続を再使用する必要がある場合は、この変数は TRUE です。新しい接続を確立する必要がある場合は、この変数は FALSE です。

値	説明
TRUE	eDirectory は、このアドレスへの接続がすでに確立されていて、再使用できると判別します。
FALSE	eDirectory は、このアドレスへの接続を持たないため、新しい接続を確立する必要があります。

- ◆ ConnectionLastUsed( 入力のみ、タイプ TIME)

ConnectionIsAlreadyOpen が TRUE の場合、ConnectionLastUsed は、この接続を使用して eDirectory から最後にパケットが送信された時点です。ConnectionIsAlreadyOpen が TRUE 以外の場合、この値は 0 になります。

値	説明
TRUE	ConnectionLastUsed は、この接続上で eDirectory が最後にパケットを送信した時点です。
FALSE	ConnectionLastUsed は 0 になります。

## NDS\_LIMBER のサンプル

eDirectory は、リンバを実行する前に、このアクティビティが許容される時間帯かどうかを WAN トラフィックマネージャに照会します。トラフィックタイプ NDS\_LIMBER には送信先アドレスがないため、NO\_ADDRESSES ポリシーが必要になります。WAN トラフィックマネージャが DONT\_SEND を返した場合は、リンバは延期され、再度スケジューリングされます。

次の変数が使用できます。

- ◆ Last( 入力のみ、タイプ TIME)  
eDirectory 起動後にリンバが最後に実行された時刻。
- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 出力のみ、タイプ INTEGER)  
リンバチェックの実行中に確立したすべての接続の有効間隔。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

- ◆ CheckEachNewOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

- ◆ CheckEachAlreadyOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します( デフォルト)。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

- ◆ Next( 出力のみ、タイプ TIME)

次のリンバチェック実行の時点。この変数を設定しないと、NDS\_LIMBER ではデフォルトが使用されます。

## NDS\_LIMBER\_OPEN のサンプル

NDS\_LIMBER\_OPEN は、対応する NDS\_LIMBER クエリの実行中に CheckEachNewOpenConnection または CheckEachAlreadyOpenConnection が 1 に設定されていた場合にのみ使用されます。このクエリは、CheckEachNewOpenConnection が 1 に設定されていて eDirectory がリンバを実行する前に新しい接続を確立する必要がある場合には、常に生成されます。このクエリは、CheckEachNewOpenConnection が 1 で eDirectory がスキーマ同期の前に新しい接続を確立する必要がある場合、または CheckEachAlreadyOpenConnection が 1 で eDirectory が既存の接続を再使用する必要がある場合には、常に生成されます。

- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 入力および出力、タイプ INTEGER)  
この接続に割り当てる有効間隔です。

値	説明
<0, 0	デフォルト有効間隔を使用( デフォルト)。
>0	この接続に割り当てる有効間隔。

- ◆ ConnectionIsAlreadyOpen( 入力のみ、タイプ BOOLEAN)

値	説明
TRUE	eDirectory は、このアドレスへの接続がすでに確立されていて、再使用できると判別します。
FALSE	eDirectory は、このアドレスへの接続を持たないため、新しい接続を確立する必要があります。



- ◆ ConnectionLastUsed( 入力のみ、タイプ TIME)

ConnectionIsAlreadyOpen が TRUE の場合、ConnectionLastUsed は、この接続を使用して DS から最後にパケットが送信された時点です。ConnectionIsAlreadyOpen が TRUE 以外の場合、この値は 0 になります。

値	説明
TRUE	ConnectionLastUsed は、この接続上で eDirectory が最後にパケットを送信した時点です。
FALSE	ConnectionLastUsed は 0 になります。

## NDS\_SCHEMA\_SYNC のサンプル

eDirectory は、スキーマを同期する前に、このアクティビティが許容される時間帯かどうかを WAN トラフィックマネージャに照会します。トラフィックタイプ NDS\_SCHEMA\_SYNC には送信先アドレスがないため、NO\_ADDRESSES ポリシーが必要になります。WAN トラフィックマネージャが DONT\_SEND を返した場合は、スキーマ同期は延期され、再度スケジューリングされます。

次の変数が使用できます。

- ◆ Last( 入力のみ、タイプ TIME)  
すべてのサーバに対してスキーマの同期が最後に正常に実行された時刻。
- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 出力のみ、タイプ INTEGER)  
スキーマの同期中に確立したすべての接続の有効間隔。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

- ◆ CheckEachNewOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

- ◆ CheckEachAlreadyOpenConnection( 出力のみ、タイプ INTEGER)

値	説明
0	WAN トラフィックマネージャを呼び出さずに成功を返します。接続は正常に続行します ( デフォルト )。
1	WAN トラフィックマネージャを呼び出します。ポリシーにより、接続を許可するかどうかが決まります。
2	WAN トラフィックマネージャを呼び出さずに ERR_CONNECTION_DENIED を返します。接続は失敗します。

## NDS\_SCHEMA\_SYNC\_OPEN のサンプル

NDS\_SCHEMA\_SYNC\_OPEN は、対応する NDS\_SCHEMA\_SYNC クエリの実行中に CheckEachNewOpenConnection または CheckEachAlreadyOpenConnection が 1 に設定されていた場合にのみ使用されます。このクエリは、CheckEachNewOpenConnection が 1 で eDirectory がスキーマ同期の前に新しい接続を確立する必要がある場合、または CheckEachAlreadyOpenConnection が 1 で eDirectory が既存の接続を再使用する必要がある場合には、常に生成されます。

- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 入力および出力、タイプ INTEGER)  
この接続に割り当てる有効間隔です。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

- ◆ ConnectionIsAlreadyOpen( 入力のみ、タイプ BOOLEAN)

値	説明
TRUE	eDirectory は、このアドレスへの接続がすでに確立されていて、再使用できると判別します。
FALSE	eDirectory は、このアドレスへの接続を持たないため、新しい接続を確立する必要があります。

- ◆ ConnectionLastUsed( 入力のみ、タイプ TIME)

ConnectionIsAlreadyOpen が TRUE の場合、ConnectionLastUsed は、この接続を使用して eDirectory から最後にパケットが送信された時点です。ConnectionIsAlreadyOpen が TRUE 以外の場合、この値は 0 になります。

値	説明
TRUE	ConnectionLastUsed は、この接続上で eDirectory が最後にパケットを送信した時点です。
FALSE	ConnectionLastUsed は 0 になります。

## NDS\_SYNC のサンプル

eDirectory がレプリカを同期する必要がある場合は、トラフィックタイプ NDS\_SYNC を使用して、WAN トラフィックマネージャへのクエリが常に生成されます。eDirectory の WAN ポリシーでは、次の変数が使用できます。

- ◆ Last( 入力のみ、タイプ TIME)  
このレプリカの同期が最後に正常に実行された時刻。
- ◆ Version( 入力のみ、タイプ INTEGER)  
eDirectory のバージョン。
- ◆ ExpirationInterval( 出力のみ、タイプ INTEGER)  
更新されたレプリカを保持するサーバへの接続の有効間隔。

値	説明
<0, 0	デフォルト有効間隔を使用 ( デフォルト )。
>0	この接続に割り当てる有効間隔。

## ONOSPOOF.WMG

このグループのポリシーでは、既存の WAN 接続のみ使用が許可されます。次の 2 つのポリシーがあります。

- ◆ Already Open, No Spoofing, NA  
バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が、既存の WAN 接続上だけに制限されます。
- ◆ Already Open, No Spoofing  
他のすべてのトラフィックが既存の WAN 接続上だけに制限されます。

すべてのトラフィックを既存の接続上だけに制限するには、両方のポリシーを適用する必要があります。

## OPNSPOOF.WMG

このグループのポリシーでは、既存の WAN 接続のみ使用が許可されますが、15 分間使用されていない接続は無効と見なされます。この場合は使用できません。次の 2 つのポリシーがあります。

- ◆ Already Open, Spoofing, NA  
このポリシーでは、バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が、15 分以内に使用されたことがある既存の WAN 接続上だけに制限されます。
- ◆ Already Open, Spoofing  
このポリシーでは、他のすべてのトラフィックが、15 分以内に使用されたことがある既存の WAN 接続上だけに制限されます。

すべてのトラフィックを 15 分以内に使用されたことがある既存の接続上だけに制限するには、両方のポリシーを適用する必要があります。

## SAMEAREA.WMG

このグループのポリシーでは、同じネットワークエリア内でのみトラフィックが許可されます。ネットワークエリアは、アドレスのネットワークセクションにより決定されます。WANトラフィックマネージャは、TCP/IP アドレスをクラス C アドレスとして解釈します(クラス C アドレスとは、アドレスの最初の 3 つのセクションが同じで同じネットワークエリアに属するものです)。IPX アドレスでは、ネットワーク部分が同じアドレスはすべて同じネットワークエリアに属すると見なされます。次の 3 つのポリシーがあります。

- ◆ Same Network Area, NA

同じネットワークエリア内で生成されるトラフィック以外の場合、バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が抑止されます。

- ◆ Same Network Area, TCPIP

同じ TCP/IP ネットワークエリア内で生成される TCP/IP トラフィック以外の場合、トラフィックが制限されます。

- ◆ Same Network Area, IXP

同じ IPX ネットワークエリア内で生成されるトラフィック以外の場合、IPX トラフィックが制限されます。

## TCPIP.WMG

このグループのポリシーでは、TCP/IP トラフィックのみ許可されます。次の 2 つのポリシーがあります。

- ◆ TCPIP, NA

TCP/IP によって生成されたトラフィック以外の場合、バックリンク、外部参照、およびログイン制限のチェック、ジャンタまたはリンバの実行、スキーマの同期が抑止されます。

- ◆ TCPIP

TCP/IP によって生成されたトラフィック以外の場合、他のすべてのトラフィックが抑止されます。

TCP/IP 以外のトラフィックをすべて抑止するには、両方のポリシーを適用する必要があります。

## TIMECOST.WMG

このグループのポリシーでは、すべてのトラフィックが午前 1 時から午前 1 時 30 分までの時間帯だけに制限されますが、同じロケーション内のサーバには連続的な対話が許可されます。このグループでは次のポリシーを使用します。すべてのポリシーを適用する必要があります。

- ◆ COSTLT20

NA およびアドレストラフィックの優先度は 40 です。

- ◆ Disallow Everything

トラフィックの送信がまったく許可されません。セレクトが 0 より大きい値を返したポリシーを WAN トラフィックマネージャが検出できない (0 個のポリシーを検出した) 場合、WAN トラフィックマネージャはデフォルトで SEND を設定します。このポリシーを使用すると、このような事態を抑止できます。

- ◆ NDS Synchronization

NDS\_SYNC トラフィックが午前 1 時と午前 1 時 30 分の間だけに制限されます。

- ◆ Start Rest.Procs, NA

すべてのプロセスを任意の時点に開始することが許可されますが、\*\_OPEN コールごとに WAN トラフィックマネージャに照会する必要があります。WAN トラフィックマネージャは、1 日 4 回 (1:00、7:00、13:00、および 19:00) の実行時刻に合わせてプロセスをスケジュールリングします。

- ◆ Start Unrest.Procs 1-1:30, NA

すべてのプロセスを午前 1 時と午前 1 時 30 分の間開始して、それ以降 WAN トラフィックマネージャに照会することなく完了するまで実行することが許可されます。プロセスは 1 日 4 回、6 時間ごとに実行されます。1:00 のプロセスはこのポリシーにより処理されますが、それ以外のプロセスは Start Rest. により処理されます。Procs, NA.

## WAN ポリシーの構成

WAN ポリシーは次の 3 つのセクションから構成されます。

- ◆ [265 ページの「宣言セクション」](#)
- ◆ [267 ページの「セクタセクション」](#)
- ◆ [268 ページの「プロバイダセクション」](#)

## 宣言セクション

ポリシーの宣言セクションには、ローカル変数およびクライアントの要求を通して入力される変数の定義が含まれています。これらの定義は、セクタセクションおよびプロバイダセクションで使用されます。これらの変数は、システム定義の変数とともに保存されます。

変数の宣言はセミコロン (;) で区切られます。同じ型の宣言が複数ある場合は、組み合わせて 1 行にすることも、複数の行に分割することもできます。行によって宣言が区別されることはありません。宣言セクションの例を次に示します。

```
REQUIRED INT R1;  
REQUIRED TIME R2;  
REQUIRED BOOLEAN R3,R4;  
REQUIRED NETADDRESS R5,R6;  
OPTIONAL INT P1 := 10;  
OPTIONAL BOOLEAN := FALSE;  
LOCAL INT L1 :=10;  
LOCAL INT L2;  
LOCAL TIME L3;  
LOCAL BOOLEAN L4 :=TRUE, L5 :=FALSE;  
LOCAL NETADDRESS L6;
```

使用される必須宣言およびオプション宣言の種類は、トラフィックタイプにより異なります。必須変数が含まれていないポリシーは実行されません。オプション宣言には、何も値が渡されない場合にデフォルトとして使用される値が指定されている必要があります。WAN トラフィックマネージャにより、すべてのトラフィックタイプで使用できるシステムシンボル (定義済み変数) が提供されます。

各宣言は次の 3 つの部分から構成されます。

- ◆ スコープ
- ◆ タイプ
- ◆ 名前とオプション値が対になっているリスト

## スコープ

有効なスコープを次の表に示します。

スコープ	説明
REQUIRED	<p>スコープが REQUIRED として定義された変数は複数のセクションで使用できますが、宣言セクションでは 1 回しか指定できません。</p> <p>REQUIRED スコープ変数に対して値を定義することはできません。この変数の値は GetWanPolicy 要求を通して入力される必要があります。</p>
OPTIONAL	<p>スコープが OPTIONAL として定義された変数は 1 つのポリシーの複数のセクションで使用できますが、宣言セクションでは 1 回しか指定できません。</p> <p>OPTIONAL スコープ変数にはデフォルト値が割り当てられます。これらの値は初期化されません。これらは、値が渡されない場合にのみ設定されます。名前とタイプの両方が一致するパラメータへ WAN ポリシー要求から新しい値が渡されない場合、ポリシーの処理では宣言で定義された値が使用されます。</p> <p>スコープが OPTIONAL として定義された変数には値を割り当てる必要があります。TIME タイプおよび NETADDRESS タイプは宣言セクションで初期化できないため、これらの変数タイプでは OPTIONAL スコープは使用しないでください。</p>
LOCAL	<p>スコープが LOCAL として定義された変数は複数のセクションで使用できますが、宣言セクションでは 1 回しか指定できません。</p> <p>LOCAL スコープ変数は特定のポリシーに対してのみ指定されます。つまり、これらの変数の値は呼び出し側クライアントには返されません。</p> <p>すべてのパラメータタイプを定義できます。ただし、TIME タイプおよび NETADDRESS タイプは宣言セクションで初期化できないため、これらのタイプには値を割り当てないでください。</p>
SYSTEM	<p>スコープが SYSTEM として定義された変数は複数のセクションで使用できますが、宣言セクションでは 1 回しか指定できません。</p>

## タイプ

有効なタイプを次の表に示します。

タイプ	説明
INT	<p>ポリシー実行を開始した GetWanPolicy 要求のトラフィックタイプを反映します。たとえば、次のポリシーでは NDS_SYNC のトラフィックタイプが指定されます。</p> <pre>IF TrafficType=NDS_SYNC THEN action END.</pre>
BOOLEAN	<p>TRUE または FALSE のいずれかの値である場合に使用します。宣言または WAN ポリシー要求で値が設定されていない場合には、値は不定です。</p>

タイプ	説明
TIME	TIME スコープ変数は、セレクトアセクションまたはプロバイダセクションで、あるいは WAN ポリシー要求から値を受け取る必要があります。宣言では TIME スコープ変数に値を割り当てないでください。
NETADDRESS	NETADDRESS スコープ変数は、セレクトアセクションまたはプロバイダセクションで値を受け取る必要があります。宣言では NETADDRESS スコープ変数に値を割り当てないでください。

宣言セクションでは、TIME タイプおよび NETADDRESS タイプに値を割り当てることはできません。TIME タイプおよび NETADDRESS タイプが値を持っていない場合、これらのタイプはセレクトアセクションまたはプロバイダセクションで値を受け取ります。宣言セクションでは、シングルタイプのみ初期化されます。

## 名前とオプション値の対

変数名は、英数字を組み合わせた任意の長さの文字列です。最初の 31 文字だけが使用されるため、変数名の最初の 31 文字は固有の文字列にする必要があります。変数名の最初の文字は英字、または定数値として解釈されるシンボルにする必要があります。

変数名では大文字と小文字が区別されます。たとえば、変数 *RI* と変数 *ri* は異なる変数として解釈されます。変数名ではアンダースコア文字 (`_`) を使用できます。

宣言内の値は、変数や式ではなく、定数でなければなりません。つまり、「LOCAL INT L2:= L3;」という宣言は許可されません。宣言セクションで変数を初期化する値は、ポリシーのセレクトアセクションおよびプロバイダセクションで変更できます。

## セレクトアセクション

ポリシーのセレクトアセクションは、キーワード `SELECTOR` で始まり、キーワード `END` で終わります。セレクトアセクションが評価されて、どのロード済みポリシーを使用するかが決定されます。

どのポリシーのウェイトが最大かを判断するために、現在ロード済みのすべてのポリシーのセレクトアセクションが実行されます。評価後、セレクトアセクションから 0 ~ 100 の間のウェイトが返されます。0 は、このポリシーを使用しないことを意味します。1 ~ 99 は、さらに大きい値が他のポリシーから返されない限り、このポリシーを使用することを意味します。100 は、このポリシーを使用することを意味します。

セレクトアセクションの結果は、`RETURN` 宣言で返されます。`RETURN` 宣言がない場合、デフォルト値の 0 が返されます。セレクトアセクションの例を次に示します。

```
SELECTOR
RETURN 49;
END
```

複数のポリシーのセレクトアセクションが評価されると、複数のポリシーから同じ値が返されることがあります。この場合、どのポリシーが選択されるかは不定です。他の条件がすべて同じである場合、サーバポリシーが WAN ポリシーに優先します。

宣言を記述する方法の詳細については、268 ページの「[ポリシーセクションで使用される構文](#)」を参照してください。268 ページの「[プロバイダセクション](#)」も参照してください。

## プロバイダセクション

プロバイダセクションは、キーワード **PROVIDER** で始まり、キーワード **END** で終わります。プロバイダセクションの主要部は宣言リストで構成されます。

この宣言リストの結果は、**SEND** または **DONT\_SEND** に対するポリシーの提案を表す値になります。

プロバイダセクションの結果は、**RETURN** 宣言で返されます。**RETURN** 宣言がない場合、デフォルト値の **SEND** が返されます。

プロバイダセクションの例を次に示します。

```
PROVIDER
RETURN SEND;
END
```

宣言を記述する方法の詳細については、[268 ページの「ポリシーセクションで使用される構文」](#)を参照してください。

## ポリシーセクションで使用される構文

WAN ポリシーのセレクトセクションおよびプロバイダセクションでは ( 宣言セクションを除く )、次のステートメントと構文を使用できます。ポリシーの宣言セクションを記述する方法の詳細については、[265 ページの「宣言セクション」](#)を参照してください。

### コメント

コメントを記述するには、行の初めに **/\*** を入力し、終わりに **\*/** を入力します。次に例を示します。

```
/* This is a comment. */
```

行の終わりで **/\*\*** に続けてコメントを記入することもできます。次に例を示します。

```
IF L2 > L3 THEN /**This is a comment.
```

### IF-THEN ステートメント

IF-THEN ステートメントを使用して、宣言ブロックを条件付きで実行します。

例:

```
IF Boolean_expression THEN declarations
END
```

```
IF Boolean_expression THEN declarations
ELSE declarations
END
```

```
IF Boolean_expression THEN declarations
ELSIF Boolean_expression THEN declarations
END
```

#### **IF Boolean\_Expression THEN**

これは IF-THEN ステートメントの最初の句です。ブール式が評価され、結果として **TRUE** または **FALSE** が返されます。**TRUE** の場合は、直後の宣言が実行されます。**FALSE** の場合は、対応する後置の **ELSE**、**ELSIF**、または **END** 宣言にジャンプします。



## ELSE

対応する IF-THEN 文または ELSIF 文の結果がすべて FALSE になる場合は、ELSE で始まる宣言が実行されます。次に例を示します。

```
IF Boolean_expression THEN statements
ELSIF Boolean_expression THEN statements
ELSIF Boolean_expression THEN statements
ELSE statements
END
```

## ELSIF Boolean\_Expression THEN

前置の IF-THEN 宣言から FALSE が返された場合、ブール式が評価されます。ELSIF 宣言が評価され、結果として TRUE または FALSE が返されます。TRUE の場合は、直後の宣言が実行されます。FALSE の場合は、対応する後置の ELSE、ELSIF、または END 宣言にジャンプします。次に例を示します。

```
IF Boolean_expression THEN statements
ELSIF Boolean_expression THEN statements
ELSIF Boolean_expression THEN statements
END
```

## END

END 宣言によって IF-THEN 構文が終了します。

## RETURN

RETURN 宣言によって、セレクトアセクションおよびプロバイダセクションの結果が返されます。

### セレクトア

セレクトアセクションでは、ポリシーのウェイトとして使用される整数が RETURN 宣言によって返されます。RETURN は 0 ~ 100 の間のポリシーウェイトを返します。0 は、このポリシーを使用しないことを意味します。1 ~ 99 は、さらに大きい値が他のポリシーから返されない限り、このポリシーを使用することを意味します。100 は、このポリシーを使用することを意味します。セレクトアセクションでは RETURN 宣言がない場合、デフォルト値の 0 が返されます。

宣言を終了するにはセミコロン (;) が必要です。次に例を示します。

```
RETURN 49;
RETURN L2;
RETURN 39+7;
```

### プロバイダ

プロバイダセクションでは、RETURN 宣言によって SEND または DONT\_SEND が返されます。RETURN 宣言がない場合、デフォルト値の SEND が返されます。

宣言を終了するにはセミコロン (;) が必要です。次に例を示します。

```
RETURN SEND;
RETURN DONT_SEND;
RETURN L1;
```

## 割り当て

割り当て宣言では、:= 文字を使用してシンボルの値が変更されます。定義済み変数またはシステム変数を最初に指定し、:= に続いて値、変数、または演算式を指定します。代入宣言の終りには、セミコロン (;) が必要です。次に例を示します。

```
variable.field:=expression; variable:=expression;
```

t1 と t2 はタイプ TIME、i1 と i2 はタイプ INTEGER、b1 と b2 はタイプ BOOLEAN の有効な割り当てです。

```
t1 := t2;
b1 := t1 < t2;
i1 := t1.mday - 15;
b2 := t2.year < 2000
```

無効な割り当ての例を示します。

```
b1 := 10 < i2 < 12;
```

(10 < i2) はブール式です。BOOLEAN を INTEGER と比較することはできません。

b1 := (10 < i2) AND (i2 < 12); を使用します。次に例を示します。

```
b2 := i1;
```

b2 は BOOLEAN で、i1 は INTEGER です。BOOLEAN を INTEGER と比較することはできません。

代わりに、b2 := i1 > 0; を使用します。

厳密なタイプのチェックが行われます。TIME変数にINTを割り当てることはできません。

## 算術演算子

割り当て宣言、RETURN 宣言、または IF 構文内で、算術演算子を使用できます。有効な演算子は次のとおりです。

- ◆ 加算 (+)
- ◆ 減算 (-)
- ◆ 除算 (/)
- ◆ 乗算 (\*)
- ◆ モジュール (MOD)

算術演算子とともに使用できるのは INT 変数タイプだけです。算術式で、TIME、NETADDRESS、または BOOLEAN 変数タイプは使用しないでください。

結果が -2147483648 ~ +2147483648 の範囲外の値になる演算、または 0 による除算は行わないでください。

## 関係演算子

IF 構文で関係演算子を使用できます。有効な演算子は次のとおりです。

- ◆ 等しい (=)
- ◆ 等しくない (<>)
- ◆ より大きい (>)

- ◆ 以上 ( $\geq$ )
- ◆ より小さい ( $<$ )
- ◆ 以下 ( $\leq$ )

関係演算子は、TIME および INT 変数タイプで使用できます。 $<$   $>$  および  $=$  は、NET ADDRESS および BOOLEAN 変数タイプでも使用できます。

## 論理演算子

有効な演算子は次のとおりです。

- ◆ AND
- ◆ OR
- ◆ NOT
- ◆ より小さい ( $<$ )
- ◆ より大きい ( $>$ )
- ◆ 等しい ( $=$ )

## ビットワイズ演算子

INT 変数タイプでビットワイズ演算子を使用して、整数値を返すことができます。有効な演算子は次のとおりです。

- ◆ BITAND
- ◆ BITOR
- ◆ BITNOT

## 複合演算

複合演算を処理するときは、次の優先規則が適用されます。同じ優先順位の演算子には、左から右へ優先順位が付けられます。優先順位は次のとおりです。

- ◆ カッコ
- ◆ 単項 (+/-)
- ◆ BITNOT
- ◆ BITAND
- ◆ BITOR
- ◆ 乗算、除算、MOD
- ◆ 加算、減算
- ◆ 関係 ( $>$ 、 $\geq$ 、 $<$ 、 $\leq$ 、 $=$ )
- ◆ NOT
- ◆ AND
- ◆ OR

優先順位が明確でない場合は、カッコを使用します。たとえば、A、B、および C が整数または変数の場合、 $A < B < C$  は許可されません。A < B では、整数値ではなくブール値が返されるので、整数 C と比較することはできません。しかし、 $(A < B) \text{ AND } (B < C)$  は文法的に正しい式です。

## PRINT

PRINT 宣言を使用して、サーバの WAN トラフィックマネージャ表示画面およびログファイルに、テキストやシンボルの値を送ることができます。

PRINT ステートメントには、リテラル文字列、シンボル名やシンボルメンバー、整数値、またはブール値などの引数をコンマで区切って、いくつでも指定できます。

リテラル文字列は二重引用符 (“ ”) で囲む必要があります。PRINT 宣言の終わりには、セミコロン (;) が必要です。次に例を示します。

```
PRINT "INT=",10,"BOOL=",TRUE,"SYM=",R1;
```

TIME および NETADDRESS 変数では、フォーマット化された PRINT 宣言が使用されます。TIME シンボルは次のよう出力されます。

```
m:d:y h:m
```

NETADDRESS 変数は次のよう出力されます。

```
Type length data
```

*type* は IP または IPX のいずれかで、*length* はバイト数、*data* は 16 進のアドレス文字列です。

# 11

## LDAP Services for Novell eDirectory について

LDAP (Lightweight Directory Access Protocol) は、クライアントアプリケーションでディレクトリ情報にアクセスするためのインターネット通信プロトコルです。LDAP は、X.500 DAP (Directory Access Protocol) に基づいていますが、従来のクライアントほど複雑ではなく、X.500 標準に基づくその他のディレクトリサービスと同時に使用することができます。

一般に、LDAP は最も単純なディレクトリアクセスプロトコルとして使用されます。

LDAP (Lightweight Directory Access Protocol) Services for Novell® eDirectory™ サーバアプリケーションを使用すると、eDirectory 内に格納されている情報に LDAP クライアントからアクセスできます。

LDAP サービスには、LDAP を通じて利用できる次のような eDirectory 機能が含まれます。

- ◆ プロビジョニング
- ◆ アカウント管理
- ◆ 認証
- ◆ 許可
- ◆ 識別情報管理
- ◆ 通知
- ◆ レポートイング
- ◆ 認定
- ◆ セグメンテーション

クライアントごとに異なるディレクトリアクセスレベルを設定して、ディレクトリにアクセスするための安全な接続を確立できます。このセキュリティメカニズムを利用すると、一般に公開するディレクトリ情報、組織内で利用する情報、および特定のグループまたは個人だけが利用できる情報を区別して管理できます。

各 LDAP クライアントで利用できるディレクトリ機能は、LDAP クライアントおよび LDAP サーバに組み込まれた機能により異なります。たとえば、LDAP Services for eDirectory を利用すると、LDAP クライアントは eDirectory データベース内のデータを読み書きできます。ただし、これには LDAP クライアントに必要な許可が与えられている必要があります。クライアントは、ディレクトリデータの読み書きが許可される場合と、読み込みしか許可されない場合があります。

一般的なクライアント機能を利用すると、クライアントから次のような処理を実行できます。

- ◆ 電子メールアドレスや電話番号など、特定の個人についての情報を検索する。
- ◆ 特定の姓または特定の文字で始まる姓を持つすべての個人の情報を検索する。
- ◆ 任意の eDirectory オブジェクトまたはエントリについての情報を検索する。

- ◆ 氏名、電子メールアドレス、勤務先電話番号、および自宅電話番号を取得する。
- ◆ 会社名および市町村名を取得する。

以降のセクションで、LDAP Services for eDirectory について説明します。

- ◆ 274 ページの「LDAP サービスの主な用語」
- ◆ 276 ページの「LDAP と eDirectory の連携について」
- ◆ 285 ページの「Linux、Solaris、AIX、または HP-UNIX 環境での LDAP ツールの使用」
- ◆ 295 ページの「拡張可能一致検索フィルタ」

LDAP サービスの詳細については、『Novell LDAP Developer Documentation(Novell LDAP 開発マニュアル) ([http://developer.novell.com/ndk/doc\\_novell\\_edirectory.htm](http://developer.novell.com/ndk/doc_novell_edirectory.htm))』を参照してください。

LDAP の詳細については、次の Web サイトを参照してください。

- ◆ ミシガン大学 (<http://www.umich.edu/~dirsvcs/ldap/ldap.html>)
- ◆ L DAP Roadmap & FAQ (<http://www.kingsmountain.com/ldapRoadmap.shtml>)
- ◆ LDAPzone.com (<http://www.ldapzone.com>)

## LDAP サービスの主な用語

### クライアントとサーバ

**LDAP クライアント** — Netscape\* Communicator\*、Internet Explorer、Novell インポート / エクスポート変換ユーティリティなどのような 1 つのアプリケーションです。

**LDAP サーバ** — nldap.nlm (NetWare® の場合)、nldap.dlm (Windows 2000/NT の場合)、libnldap.so (Linux、Solaris および AIX システムの場合)、または libnldap.sl (HP-UX システムの場合) が稼動するサーバ。

### オブジェクト

**LDAP グループオブジェクト** — LDAP サーバで Novell LDAP プロパティの設定と管理を行います。

このオブジェクトは eDirectory のインストール時に作成されます。LDAP グループオブジェクトには、複数の LDAP サーバ間で共有できる便利な設定情報が含まれています。

**LDAPサーバオブジェクト** — LDAPクライアントによる情報へのアクセスおよび使用方法の設定と管理を Novell LDAP サーバで行います。

このオブジェクトは eDirectory のインストール時に作成されます。LDAP サーバオブジェクトとは、サーバ固有の設定データのことで、

次の図は、Novell iManager の LDAP サーバオブジェクトを表したものです。



## 参照

**参照** — LDAP サーバが LDAP クライアントに送信するメッセージです。このサーバからは結果がすべて提供されず、他の LDAP サーバにまだデータがある可能性をクライアントに通知します。

参照には、操作を続行するのに必要な情報がすべて含まれます。

シナリオ: LDAP クライアントが LDAP サーバにリクエストを送信しますが、サーバは操作のターゲットエントリをローカルで見つけることができません。その場合、LDAP サーバはパーティションおよび他のサーバに関して所有する知識参照を使用して、そのエントリについてより多くの知識を持つ別のサーバを特定します。LDAP サーバは参照情報をクライアントに送信します。

クライアントは識別されたサーバに対する新しい LDAP 接続を確立し、操作を再試行します。

参照には次の利点があります。

- ◆ LDAP クライアントが操作を制御し続けます。

常に状況を把握することにより、クライアントはよりの確な判断ができ、ユーザにフィードバックを返すことができます。また、参照を検索しない選択をしたり、検索の前にユーザに確認メッセージを表示することもできます。

- ◆ 多くの場合、参照の方がチェーンよりもリソースを効果的に使用できます。

チェーンでは、エントリの多い検索操作がリクエストされると、ネットワーク全体に 2 度送信される場合があります。1 度目はデータのあるサーバからチェーンを実行するサーバへの送信、2 度目はチェーンを実行するサーバからクライアントへの送信です。

参照では、クライアントはデータのあるサーバから、1 回の送信で直接データを受け取ることができます。

- ◆ エントリの格納場所がわかっている場合、クライアントは直接データのあるサーバにアクセスすることができます。

チェーンでは、クライアントは詳細を見ることはできません。エントリの格納場所がわからない場合、クライアントが直接データのあるサーバにアクセスすることはできません。

参照には次の欠点があります。

- ◆ クライアントが参照とその検索方法を認識できる能力が必要です。
- ◆ LDAP v2 クライアントでは参照結果が認識されないか、認識に古い非標準の方法が使用されます。
- ◆ すべての eDirectory パーティションに LDAP サーバのサービスが適用されていなくてはなりません。

適用されていない場合、参照結果をパーティションのデータに送信できません。

**上方参照** — 通信中のサーバのデータよりもツリーの高い位置にあるデータを持つサーバを参照することです。324 ページの「[上方参照を設定する](#)」を参照してください。

上方参照では、マルチベンダツリーで、上方または隣接した非 eDirectory パーティションにあるオブジェクトに関するリクエストが処理されます。

eDirectory サーバがこのタイプのツリーに含まれるようにするには、eDirectory は「非委任」とマークされたパーティション内で、階層データを上方に保持するようにします。

非委任領域のオブジェクトは、正しい DN 階層を構築するのに必要なエントリのみから構成されます。これらのエントリは、X.500 の “Glue” エントリに類似しています。

eDirectory では、知識情報を LDAP 参照データの形式で非委任領域に配置できます。この情報は、LDAP クライアントに参照を戻すのに使用されます。

LDAP 操作が eDirectory ツリーの非委任領域で実行されると、LDAP サーバは正しい参照データを検索し、クライアントに参照を戻します。

**チェーン** — サーバベースのネームレゾリューションプロトコル。

LDAP クライアントは LDAP サーバにリクエストを送信しますが、LDAP サーバは操作のターゲットエントリをローカルで見つけることができません。LDAP サーバ (サーバ A) は eDirectory ツリーのパーティションおよび他のサーバに関して所有する知識参照を使用して、DN についてより多くの知識を持つ、別の LDAP サーバ (サーバ B) を特定します。LDAP サーバ A は、特定された LDAP サーバ B と通信します。

必要に応じ、サーバ A がエントリのレプリカを持つサーバに接続するまでこの処理が続けられます。その後、eDirectory は詳細をすべて処理し、操作を完了します。サーバ間の処理はクライアントには表示されないため、クライアントは最初のサーバ A がリクエストを処理したものと判断します。

LDAP サーバにチェーンを使用した場合、次の利点があります。

- ◆ ネームレゾリューションの詳細をすべてクライアントから見えなくします。
- ◆ 自動で再認証を行います。
- ◆ クライアントのプロキシの役割を果たします。
- ◆ eDirectory ツリーに LDAP サービスをサポートしないサーバがあっても、シームレスに実行されます。

一方、チェーンには次の欠点があります。

- ◆ チェーンを使用して名前解決中には、サーバからのフィードバックがなくクライアントが待機する必要がある場合があります。
- ◆ LDAP サーバが、WAN リンク経由で多くのエントリを送信するよう要求された場合、処理に長時間かかることがあります。
- ◆ ほぼ同じ処理能力を持つサーバがいくつかある場合、別々のサーバが 2 つの要求を同じエントリ上で処理してしまうことがあります。

eDirectory は、サーバを接続コスト順にソートしようとします。負荷分散のため、dDirectory は一番コストの低いサーバの中からランダムに選択を行います。

## LDAP と eDirectory の連携について

このセクションでは次について説明します。

- ◆ [277 ページの「LDAP から eDirectory に接続する」](#)
- ◆ [280 ページの「クラスと属性のマッピング」](#)
- ◆ [283 ページの「非標準スキーマ出力を有効にする」](#)
- ◆ [284 ページの「構文の相違」](#)
- ◆ [285 ページの「サポートされる Novell LDAP コントロールおよび拡張」](#)



## LDAP から eDirectory に接続する

すべての LDAP クライアントが、次のいずれかのユーザタイプで Novell eDirectory にバインド (接続) されます。

- ◆ [Public] ユーザ (匿名バインド)
- ◆ プロキシユーザ (プロキシユーザ匿名バインド)
- ◆ NDS または eDirectory ユーザ (NDS ユーザバインド)

ユーザの認証に使用されるバインドタイプにより、LDAP クライアントがアクセスできる内容が決定されます。LDAP クライアントは、作成した要求をディレクトリに送信することにより、ディレクトリにアクセスします。LDAP クライアントが LDAP Services for eDirectory を通じて要求を送信した場合、eDirectory は、その中から LDAP クライアントが適切なアクセス権を持つ属性の要求だけを処理します。

たとえば LDAP クライアントが、読み込み権が必要なある属性値を要求したものの、その属性についてユーザに許可されているのが比較権だけである場合、この要求は拒否されます。

標準ログイン制限とパスワード制限は引き続き適用されます。ただし、制限はすべて LDAP の実行場所と関係します。時刻およびアドレス制限も適用されますが、アドレス制限は eDirectory ログインが実行された場所 (この場合は LDAP サーバ) を基準に決定されます。

### [Public] ユーザとして接続する

匿名バインドは、ユーザ名またはパスワードを使用しない接続です。サービスでプロキシユーザの使用が設定されていない場合、名前とパスワードが定義されていない LDAP クライアントで LDAP Services for eDirectory にバインドすると、ユーザは eDirectory に [Public] ユーザとして認証されます。

[Public] ユーザとは、非認証の eDirectory ユーザのことです。デフォルトでは、[Public] ユーザには eDirectory ツリー内のオブジェクトのブラウズ権が割り当てられます。[Public] ユーザ用のデフォルトブラウズ権では、Directory オブジェクトを参照することはできませんが、ほとんどのオブジェクト属性にアクセスすることはできません。

多くの場合、LDAP クライアントは、デフォルトの [Public] 権だけでは不十分です。[Public] の権利は変更できますが、変更した権利はすべてのユーザに対して許可されることとなります。この問題を解決するために、プロキシユーザ匿名バインドを使用することをお勧めします。詳細については、[277 ページの「プロキシユーザとして接続する」](#)を参照してください。

[Public] ユーザによるオブジェクト属性へのアクセスを許可するには、[Public] ユーザを該当する (1 つまたは複数の) コンテナのトラスティに設定し、適切なオブジェクト権および属性権を割り当てる必要があります。

### プロキシユーザとして接続する



プロキシユーザ匿名バインドは、eDirectory ユーザ名にリンクされた匿名接続です。プロトコルでプロキシユーザの使用が設定されている場合、LDAP クライアントが LDAP Services for eDirectory に匿名でバインドすると、ユーザは eDirectory によりプロキシユーザとして認証されます。LDAP Services for eDirectory と eDirectory の両方でユーザ名が設定されます。

通常、匿名バインドには LDAP のポート 389 が使用されます。ただし、ポートはインストール時に手動で変更することができます。


次に、プロキシユーザ匿名バインドの概念について説明します。

- ◆ 匿名バインドを経由する LDAP クライアントアクセスは、すべてプロキシユーザオブジェクトを通して割り当てられます。
- ◆ 匿名バインドでは LDAP クライアントからパスワードが提供されないため、プロキシユーザに null パスワードを適用し、パスワード変更間隔などのパスワード制限を設定しないようにします。パスワードを強制的に有効期限切れにしたり、プロキシユーザにパスワードの変更を許可したりしないでください。
- ◆ プロキシユーザオブジェクトのアドレス制限を設定すると、ユーザがログインできるロケーションを制限できます。
- ◆ eDirectory 内にプロキシユーザオブジェクトを作成し、公開する eDirectory オブジェクトに対する権利を割り当てる必要があります。デフォルトのユーザ権では、特定のオブジェクトと属性だけに対する読み込みアクセス権が許可されます。アクセスする必要がある各サブツリー内のすべてのオブジェクトと属性に対するプロキシユーザ読み込み権および検索権を割り当てます。
- ◆ LDAP Services for eDirectory を設定する LDAP グループオブジェクトの [全般] ページで、プロキシオブジェクトを有効化する必要があります。このため、1つの LDAP グループ内のすべてのサーバに対し、プロキシユーザオブジェクトは 1つしか作成できません。詳細については、304 ページの「LDAP オブジェクトを環境設定する」を参照してください。
- ◆ プロキシユーザオブジェクトに対し、すべてのプロパティ(デフォルト)か、選択したプロパティの権利を許可できます。

プロキシユーザに対して選択したプロパティの権利だけを許可するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [権利] > [トラスティの変更] の順にクリックします。
- 3 プロキシユーザが権利を持つ最上部のコンテナの名前とコンテキストを指定するか、 をクリックし、問題のコンテナを参照して [OK] をクリックします。
- 4 [トラスティの変更] 画面で [トラスティの追加] をクリックします。
- 5 プロキシユーザのオブジェクトを参照し、[OK] をクリックします。
- 6 追加したプロキシユーザの左側にある [割り当てられた権利] をクリックします。
- 7 [すべての属性権] および [エントリ権] チェックボックスをオンにし、[プロパティの削除] をクリックします。
- 8 [プロパティの追加] をクリックし、[スキーマ内のすべてのプロパティを表示する] チェックボックスをオンにします。
- 9 メールボックス(リストの小文字のセクション)や役職など、プロキシユーザが継承可能な権利を選択し、[OK] をクリックします。  
その他の継承可能な権利を追加する場合は、手順 9 ~ 10 を繰り返します。
- 10 [完了] をクリックし、[OK] をクリックします。

プロキシユーザ匿名バインドを実装するには、eDirectory 内にプロキシユーザオブジェクトを作成し、そのユーザに適切な権利を割り当てる必要があります。アクセスする必要がある各サブツリー内のすべてのオブジェクトと属性に対するプロキシユーザ読み込み権および検索権を割り当てます。同じプロキシユーザ名を指定して、LDAP Services for eDirectory 内でプロキシオブジェクトを有効化する必要があります。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。

- 2 [LDAP] > [LDAP の概要] の順にクリックします。
- 3 設定する LDAP グループオブジェクトの名前をクリックします。
- 4 [プロキシユーザ] フィールドに eDirectory ユーザオブジェクトの名前とコンテキストを指定します。
- 5 [適用] をクリックし、[OK] をクリックします。

### UNIX の ldapconfig を使用する

たとえば、LDAP Search Referral Usage で LDAP サーバによる LDAP 参照の処理方法を指定します。

- 1 システムプロンプトで、次のコマンドを入力します。  

```
ldapconfig -s "LDAP:otherReferralUsage=1"
```
- 2 ユーザ FDN(完全識別 eDirectory ユーザ名) とパスワードを入力します。

### NDS または eDirectory ユーザとして接続する

eDirectory ユーザバインドは、LDAP クライアントが完全な eDirectory ユーザ名とパスワードを使用して確立する接続です。eDirectory ユーザバインドは eDirectory により認証されます。LDAP クライアントは、その eDirectory ユーザにアクセスが許可されているすべての情報にアクセスできます。

次に、eDirectory ユーザバインドに関する重要な概念について説明します。

- ◆ eDirectory ユーザバインドは、LDAP クライアントに入力されたユーザ名とパスワードを使用して eDirectory により認証されます。
- ◆ LDAP クライアントアクセスに使用する eDirectory ユーザ名とパスワードは、NetWare クライアントが eDirectory にアクセスする場合も使用できます。
- ◆ 非 TLS 接続では、eDirectory パスワードは LDAP クライアントと LDAP Services for eDirectory の間の経路をクリアテキストデータとして転送されます。
- ◆ クリアテキストパスワードが無効に設定されている場合は、非 TLS 接続上で転送されたユーザ名またはパスワードを含む eDirectory バインド要求は、すべて拒否されます。
- ◆ eDirectory ユーザパスワードの有効期限が切れた場合、そのユーザの eDirectory バインド要求は拒否されます。

### LDAP クライアントに eDirectory 権を割り当てる

- 1 LDAP クライアントが eDirectory にアクセスするときに使用するユーザ名のタイプを決定します。
  - ◆ [Public] ユーザ (匿名バインド)
  - ◆ プロキシユーザ (プロキシユーザ匿名バインド)
  - ◆ NDS ユーザ (NDS ユーザバインド)

詳細については、[277 ページの「LDAP から eDirectory に接続する」](#)を参照してください。

- 2 ユーザが1つのプロキシユーザまたは複数の eDirectory ユーザ名で LDAP にアクセスする場合、iManager を使用して、eDirectory 内または LDAP でこれらのユーザ名を作成します。
- 3 LDAP クライアントが使用するユーザ名に、適切な eDirectory 権を割り当てます。

ほとんどのユーザに割り当てられるデフォルトの権利では、ユーザ自身が持つオブジェクト以外にはアクセスできません。別のオブジェクトやその属性にアクセスするには、eDirectory で割り当てられた権利を変更する必要があります。

LDAP クライアントから eDirectory オブジェクトおよび属性へのアクセスが要求されると、eDirectory は、LDAP クライアントの eDirectory 識別情報に基づいて要求を受諾または拒否します。識別情報はバインド時に設定されます。

## クラスと属性のマッピング

クラスとは、ディレクトリ内のオブジェクトのタイプ (ユーザ、サーバ、グループなど) です。属性とは、特定のオブジェクトについての追加情報を定義するディレクトリ要素です。たとえば、ユーザオブジェクト属性にはユーザの姓、電話番号などがあります。


スキーマとは、ディレクトリで使用できるクラスと属性、およびディレクトリ構造 (クラス間の相互関係) を定義する一連の規則です。LDAP ディレクトリと eDirectory ディレクトリのスキーマが異なる場合は、LDAP クラスと属性を、適切な eDirectory オブジェクトと属性へマッピングしなければならない場合があります。これらのマッピングで、LDAP スキーマから eDirectory スキーマへの名前の変換を定義します。

LDAP Services for eDirectory にはデフォルトマッピングがあります。LDAP クラスおよび属性と、eDirectory オブジェクトタイプおよびプロパティとの対応関係は多くの場合論理的で、直観的に理解できます。ただし、実装時の条件によっては、クラスと属性のマッピングを再設定する必要が生じることもあります。

ほとんどの場合、LDAP クラスと eDirectory オブジェクトタイプとの間のマッピングは、一対一の対応関係です。ただし、LDAP スキーマでは、同じ属性を意味する CN および共通名のような別名もサポートされています。

## LDAP グループ属性をマッピングする

デフォルトの LDAP Services for eDirectory 環境設定には、定義済みのクラスと属性のマッピングが保存されています。これは、LDAP 属性のサブセットから eDirectory 属性のサブセットへのマッピングです。デフォルト環境設定でマッピングされていない属性には、自動生成されたマッピングが割り当てられます。スキーマ名がスペースまたはコロンを含まない有効な LDAP 名である場合は、マッピングは必要ありません。クラスおよび属性のマッピングを調べて、必要に応じて再設定します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] > [LDAP グループの表示] の順にクリックします。
- 3 LDAP グループオブジェクトをクリックし、[属性マップ] をクリックします。
- 4 必要に応じて属性を追加、削除、または変更します。


LDAP 属性の種類によっては別名 (CN および共通名など) が存在する場合があるため、複数の LDAP 属性を対応する 1 つの eDirectory 属性名にマッピングする必要があります。LDAP Services for eDirectory が LDAP 属性情報を返す場合、リスト内で検出された最初の一致する属性の値が返されます。

複数の LDAP 属性を 1 つの eDirectory 属性にマッピングする場合は、属性の順序に意味があります。リスト内の順序を変更することにより、属性の優先度を変更できます。

- 5 [適用] をクリックし、[OK] をクリックします。

## LDAP グループクラスをマッピングする

LDAP クライアントが LDAP サーバに LDAP クラス情報を要求すると、サーバは対応する eDirectory クラス情報を返します。デフォルトの LDAP Services for eDirectory 環境設定には、定義済みのクラスと属性のマッピングが保存されています。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] の順にクリックします。
- 3 LDAP グループオブジェクトをクリックし、[クラスマップ] をクリックします。
- 4 必要に応じてクラスを追加、削除、または変更します。

デフォルトの LDAP Services for eDirectory 環境設定には、定義済みのクラスと属性のマッピングが保存されています。これは、LDAP クラスと属性のサブセットから eDirectory クラスと属性のサブセットへのマッピングです。デフォルト環境設定でマッピングされていない属性またはクラスには、自動生成されたマッピングが割り当てられます。

スキーマ名がスペースまたはコロンを含まない有効な LDAP 名である場合は、マッピングは必要ありません。クラスおよび属性のマッピングを調べて、必要に応じて再設定します。

- 5 [適用] をクリックし、[OK] をクリックします。

## LDAP クラスと属性をマッピングする

LDAP ディレクトリと eDirectory ディレクトリのスキーマは異なるため、LDAP クラスと属性を、適切な eDirectory オブジェクトと属性へマッピングする必要があります。これらのマッピングで、LDAP スキーマから eDirectory スキーマへ名前の変換を定義します。

有効な LDAP スキーマ名であれば、スキーマエントリに対する LDAP スキーママッピングは必要ありません。LDAP では、スキーマ名で使用できる文字は英数字とハイフン記号 (-) だけです。LDAP スキーマ名ではスペースは使用できません。

スキーマを LDAP の外部に拡張する場合、.sch ファイルなど LDAP の外部にスキーマを拡張した後でオブジェクト ID による検索を確実に実行するには、LDAP サーバ環境設定をリフレッシュする必要があります。

## 多対一マッピング

eDirectory から LDAP をサポートするために、LDAP Services は、( ディレクトリサービスレベルではなく ) プロトコルレベルのマッピングを使用して、LDAP と eDirectory の間で属性とクラスを変換します。したがって、2 つの LDAP クラスまたは属性を同じ eDirectory クラスまたは属性にマッピングできます。

たとえば、LDAP を使用して Cn を作成し、CommonName=Value を検索すると、Cn と属性値が同じ可能性のある commonName が返されます。

すべての属性を要求すると、そのクラスのマッピングリストの最初にある属性が返されます。名前属性を要求すると、正しい名前が返されます。

## 多対一クラスマッピング

LDAP クラス名	eDirectory クラス名
alias aliasObject	Alias
groupOfNames groupOfUniqueNames group	Group
mailGroup rfc822mailgroup	NSCP:mailGroup1

## 多対一属性マッピング

LDAP 属性名	eDirectory 属性名
c countryName	C
cn commonName	CN
uid userId	uniqueID
description multiLineDescription	Description
l localityname	L
member uniqueMember	Member
o organizationname	O
ou organizationalUnitName	OU
sn surname	Surname
st stateOrProvinceName	S
certificateRevocationList;binary certificateRevocationList	certificateRevocationList
authorityRevocationList;binary authorityRevocationList	authorityRevocationList
deltaRevocationList;binary deltaRevocationList	deltaRevocationList
cACertificate;binary cACertificate	cACertificate

LDAP 属性名	eDirectory 属性名
crossCertificatePair;binary crossCertificatePair	crossCertificatePair
userCertificate;binary userCertificate	userCertificate

注：;binary の付いた属性はセキュリティに関連しています。アプリケーションに必要な名前が ;binary を付けて取得される場合は、マッピングテーブルから属性を適用します。名前を ;binary を付けずに取得する場合は、マッピングの順序を変更できます。

## 非標準スキーマ出力を有効にする

eDirectory には、互換モードスイッチがあります。この機能により、非標準スキーマ出力が使用できるため、現行の ADSI クライアントおよび従来の Netscape\* クライアントでスキーマを読み込むことができます。このスイッチは、LDAP サーバオブジェクト内の属性を設定することにより実装されます。属性名は nonStdClientSchemaCompatMode です。通常の場合、LDAP サーバオブジェクトはサーバオブジェクトと同じコンテナ内にあります。


非標準出力は、LDAP 用の現行 IETF 規格には適合しませんが、現行バージョンの ADSI クライアントおよび従来の Netscape クライアントでは正常に処理できます。

非標準出力の出力形式は次のとおりです。

- ◆ SYNTAX OID は一重引用符で囲まれます。
- ◆ 上限は出力されません。
- ◆ X- オプションは出力されません。
- ◆ 複数の名前が存在する場合は、最初に検出された名前だけが出力されます。
- ◆ OID が定義されていない属性やクラスは、それぞれ小文字で「attributename-oid」、「classname-oid」と出力されます。
- ◆ 名前にハイフンが含まれていて OID が定義されていない属性またはクラスは出力されません。

OID またはオブジェクト識別子は、自身の属性または objectclass を LDAP サーバに追加するのに必要なオクテット数値の文字列です。

非標準スキーマ出力を有効化するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
  - 2 [LDAP] > [LDAP の概要] の順にクリックします。
  - 3 [LDAP サーバの表示] をクリックし、LDAP サーバオブジェクトをクリックします。
  - 4 [検索] をクリックし、[古い ADSI および Netscape スキーマ出力を有効にする] をクリックします。
- 非標準の出力は、現在 IETF が LDAP について定義している規格に準拠していませんが、現在の ADSI クライアントおよび以前の Netscape クライアントでは動作します。
- 5 [適用]、[情報] の順にクリックし、[リフレッシュ] をクリックします。

## 構文の相違

LDAP と eDirectory では使用される構文が異なります。次のような重要な相違点があります。

- ◆ 284 ページの「コンマ」
- ◆ 284 ページの「タイプ付きの名前」
- ◆ 284 ページの「エスケープ文字」
- ◆ 284 ページの「複数のネーミング属性」

### コンマ

LDAP では、区切り記号としてピリオドではなくコンマを使用します。たとえば、eDirectory の識別名 (完全名) は次のように記述します。

```
CN=JANEBOU=MKTGO=EMA
```

LDAP 構文を使用すると、同じ識別名は次のようになります。

```
CN=JANEBOU=MKTGO=EMA
```

また、次は別の LDAP 識別名の例です。

```
CN=Bill Williams,OU=PR,O=Bella Notte Corp
```

```
CN=Susan Jones,OU=Humanities,O=University College London,C=GB
```

### タイプ付きの名前

eDirectory では、タイプなしの名前 (.JOHN.MARKETING.ABCCORP) とタイプ付きの名前 (CN=JOHN.OU=MARKETINGO=ABCCORP) の両方を使用します。LDAP では、区切り記号としてコンマを使用したタイプ付きの名前 (CN=JOHN,OU=MARKETING,O=ABCCORP) だけを使用します。

### エスケープ文字

LDAP 識別名では、エスケープ文字として円記号 (¥) を使用します。1 つの円記号とプラス記号 (+) またはコンマ (,) を指定すると、識別名を拡張できます。

次に例を示します。

```
CN=Pralines\+Cream,OU=Flavors,O=MFG (CN is Pralines+Cream)
```

```
CN=D. Cardinal,O=Lionel\,Turner and Kaye,C=US (O は Lionel、Turner、および Kaye)
```

詳細については、Internet Engineering Task Force の RFC 232 (<http://www.ietf.org/rfc/rfc2253.txt?number=2253>) を参照してください。

### 複数のネーミング属性

オブジェクトは、スキーマ内の複数のネーミング属性を使用して定義できます。LDAP と eDirectory のユーザオブジェクトには、いずれも CN と UID の 2 つのネーミング属性があります。識別名の中のネーミング属性は、プラス記号 (+) で区切ります。属性に明示的なラベルが付いていない場合は、スキーマによりそれぞれの文字列に対応する属性が決定されます (eDirectory と LDAP の両方で、最初の文字列は CN、次の文字列は UID になります)。識別名の中の各部分に手動でラベルを付けると、ネーミング属性の順序を変更できます。



2つの相対識別名の例を次に示します。

Smith (CN は Smith CN=Smith)

Smith+Lisa (CN は Smith、OU は Lisa CN=Smith UID=Lisa)

2つの相対識別名 (Smith と Smith+Lisa) は、2つの異なる相対識別名によって参照されるため、同じコンテキスト内に共存することができます。

## サポートされる Novell LDAP コントロールおよび拡張

LDAP クライアントと LDAP サーバは、LDAP 3 プロトコルを使用することにより、コントロールと拡張を適用して LDAP 操作を拡張できます。コントロールと拡張を使用することによって、要求や応答の一部として追加情報を指定できます。拡張された各操作は、自身の属性または `objectclass` を LDAP サーバに追加するのに必要な、オクテット数値の文字列であるオブジェクト識別子 (OID) により識別されます。LDAP クライアントは、実行したい拡張操作の OID およびその拡張操作に固有なデータを指定した拡張操作要求を送信できます。LDAP サーバはこの要求を受信すると、拡張操作を実行し、OID と追加データが設定された応答をクライアントに送信します。

たとえば、クライアントがサーバに検索要求を送信するとき、ソートを指定するコントロールをこの要求に入れることができます。サーバはこの検索要求を受け取ると、検索結果をソートしてから、その結果をクライアントに戻します。コントロールはサーバからクライアントに送ることもできます。たとえば、サーバは、クライアントにパスワード期限切れを通知する認証要求のコントロールを送ることができます。

デフォルトでは、起動直後の eDirectory LDAP サーバは、すべてのシステム拡張ならびに選択されたオプション拡張およびコントロールをロードできる状態にあります。オプション拡張に対応する LDAP サーバオブジェクトの `extensionInfo` 属性により、システム管理者は、オプション拡張およびコントロールの選択および選択解除ができます。

拡張操作を有効にするため、LDAP 3 プロトコルはルート DSE 内の `supportedControl` 属性および `supportedExtension` 属性に含まれる、サポートされているコントロールと拡張のリストをサーバに要求します。ルート DSE (DSE は DSA (Directory System Agent) Specific (固有) Entry (エントリ) の略) とは、ディレクトリ情報ツリー (DIT) のルートにあるエントリです。詳細については、331 ページの「LDAP サーバの情報を取得する」を参照してください。

サポートされている LDAP コントロールと拡張のリストについては、『*LDAP and NDS Integration Guide (LDAP および NDS 統合ガイド)*』の「LDAP Controls ([http://developer.novell.com/ndk/doc/ldapover/ldap\\_enu/data/cchbehhc.html](http://developer.novell.com/ndk/doc/ldapover/ldap_enu/data/cchbehhc.html))」と「LDAP Extensions ([http://developer.novell.com/ndk/doc/ldapover/ldap\\_enu/data/a6ik7oi.html](http://developer.novell.com/ndk/doc/ldapover/ldap_enu/data/a6ik7oi.html))」を参照してください。

## Linux、Solaris、AIX、または HP-UNIX 環境での LDAP ツールの使用

eDirectory には次の LDAP ツールが含まれており、これを使用して LDAP ディレクトリサーバを管理できます。これらのツールは `/usr/ldaptools/bin` (ice のみ `/usr/bin`) に格納されています。

ツール	説明
ice	エントリをファイルから LDAP ディレクトリにインポートし、ファイルのディレクトリ内のエントリを変更し、エントリをファイルにエクスポートし、ファイルの属性とクラス定義を追加します。

ツール	説明
ldapadd	LDAP ディレクトリに新しいエントリを追加します。
ldapdelete	LDAP ディレクトリサーバからエントリを削除します。ldapdelete ツールは、LDAP サーバとの接続を開始し、エントリのバインドと削除を行います。
ldapmodify	LDAP サーバとの接続を開始し、エントリのバインド、変更、追加を行います。
ldapmodrdn	LDAP ディレクトリサーバ内のエントリの相対識別名 (RDN) を変更します。LDAP サーバとの接続を開始し、エントリの RDN のバインドと変更を行います。
ldapsearch	LDAP ディレクトリサーバでエントリを検索します。LDAP サーバとの接続を開始し、バインドを行い、指定されたフィルタを使用して検索を実行します。フィルタは、RFC 2254 ( <a href="http://www.ietf.org/rfc/rfc2254.txt">http://www.ietf.org/rfc/rfc2254.txt</a> ) で定義された LDAP フィルタの文字列表現に準拠している必要があります。
ndsindex	インデックスの作成、一覧表示、一時停止、再開、削除を行います。

詳細については、『*LDAP Libraries for C Guide (LDAP Libraries for C ガイド)*』の「[LDAP Tools \(http://developer.novell.com/ndk/doc/cldap/ltoolenu/data/hevgtl7k.html\)](http://developer.novell.com/ndk/doc/cldap/ltoolenu/data/hevgtl7k.html)」を参照してください。

LDAP ツールを安全に実行するには、82 ページの「[Linux、Solaris、AIX、および HP-UX システムでの eDirectory 操作に関するセキュリティを確保する](#)」を参照し、安全に eDirectory と LDAP を接続するためのコマンドラインによる LDAP 操作に DER ファイルを指定します。

## LDAP ツール

LDAP ユーティリティは、エントリの削除、変更、追加、スキーマの拡張、相対識別名の変更、エントリの新規コンテナへの異動、検索インデックスの作成、検索の実行に使用できます。

### ldapadd

ldapadd ユーティリティを使用して、新しいエントリを追加します。構文は次のとおりです。

```
ldapadd [-c] [-C] [-l] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W ]| [-w passwd]] [-h ldaphost] [-p ldapport] [-P version] [-Z[Z]] [-f file]
```

**注:** NetWare サーバでは、このユーティリティは ladd と呼ばれます。

-f オプションを指定すると、ldapadd により変更がファイルから読み出されます。-f オプションを指定しない場合、ldapadd は変更を stdin から読み出します。

**ヒント:** ldap ユーティリティからの出力は stdout に送られます。出力が表示される前にユーティリティが存在する場合、出力はファイルにリダイレクトされます。例 :ldapadd [options] > out.txt

オプション	説明
-a	新しいエントリを追加します。ldapmodify のデフォルトは、既存エントリの変更です。ldapadd として呼び出されると、このフラグは常にオンになります。
-r	デフォルトでは既存の値を置換します。
-c	連続操作モード。エラーが通知されても、ldapmodify は変更動作を継続します。デフォルトでは、エラーが通知されると終了します。
-f file	エントリ情報を標準入力ではなく LDIF ファイルから読み出します。レコードの最大長は 4096 行です。
-F	replica: で始まる入力行の内容に関わりなく、すべての変更を無条件で適用します ( デフォルトでは、replica: 行は使用中の LDAP サーバホストおよびポートと比較され、repllog レコードを実際に適用するかどうか決定します )。

### すべての LDAP ツールの共通オプション

すべての LDAP ツールに共通するオプションがいくつかあります。次の表は、これらのオプションを表示したものです。

オプション	説明
-C	次の参照を有効にします。(匿名バインド)
-d debuglevel	LDAP デバッグレベルを debuglevel に設定します。このオプションを有効にするには、ldapmodify をコンパイルするときに LDAP_DEBUG の定義が必要です。
-D binddn	binddn を使用して LDAP ディレクトリにバインドします。binddn には、RFC 1779 に定義されている文字列表現の DN を指定します。
-e key filename	SSL バインドのため、ファイル名の確認を行います。
-f file	ファイルから行を読み取り、1 行ごとに LDAP 検索を実行します。この場合、コマンドラインに指定されたフィルタは、%s が最初に出現した個所がファイルに指定された行で置換されるというパターンとして機能します。1 つのハイフン (-) 文字がファイルとして指定された場合には、標準入力から行が読み取られます。
-h ldaphost	LDAP サーバの実行場所となっている代替ホストを指定します。
-l limit	接続タイムアウト ( 秒 ) を指定します。
-M	Manage DSA IT コントロールを有効にします。(非致命的)
-MM	Manage DSA IT コントロールを有効にします。(致命的)
-n	完了した場合の結果を表示しますが、実際にはエントリを変更しません。-v と組み合わせて使用すると、デバッグ時に便利です。
-p ldapport	LDAP サーバがリスンしている代替 TCP™ ポートを指定します。
-P version	LDAP のバージョン (2 または 3) を指定します。
-v	冗長モードが設定され、多くの診断メッセージが標準出力に書き込まれます。

オプション	説明
-w <i>passwd</i>	簡易認証のパスワードとして <i>passwd</i> を使用します。
-W	簡易認証のプロンプトです。コマンドラインにパスワードを指定するかわりに、このオプションが使用されます。
-Z	<p>操作をバインドして実行する前に、TLS を開始します。TLS を開始する操作の途中でエラーが発生すると、エラーは無視され操作は続行されます。エラーが発生した場合に操作を中止するには、このオプションの代わりに -zz オプションの使用をお勧めします。</p> <p>このオプションでポートが指定されている場合、そのポートはクリアテキスト接続を受信する必要があります。</p> <p>サーバの識別情報を確認するには、このオプションを -e オプションと組み合わせて使用し、サーバ証明書ファイルを指定する必要があります。TLS を開始すると、これによりサーバのルート認証局証明書が確認されます。-e オプションが指定されていない場合、サーバからのすべての証明書が許可されます。</p>
-ZZ	<p>操作をバインドして実行する前に、TLS を開始します。TLS を開始する操作の途中でエラーが発生すると、操作は中止されます。</p> <p>このオプションでポートが指定されている場合、そのポートはクリアテキスト接続を受信する必要があります。</p> <p>サーバの識別情報を確認するには、このオプションを -e オプションと組み合わせて使用し、サーバ証明書ファイルを指定する必要があります。TLS を開始すると、これによりサーバのルート認証局証明書が確認されます。-e オプションが指定されていない場合、サーバからのすべての証明書が許可されます。</p>

## 例

/tmp/entrymods ファイルが存在すると仮定すると、次のような内容になります。

```
dn: cn=Modify Me, o=University of Michigan, c=US
changetype: modify
replace: mail
mail: modme@terminator.rs.itd.umich.edu
-
add: title
title: Manager
-
add: jpegPhoto
jpegPhoto: /tmp/modme.jpeg
-
delete: description
-
```

この場合、コマンド「`ldapmodify -b -r -f/tmp/entrymods`」は、コンテンツ「Modify Me entry」のメール属性を「`modme@terminator.rs.itd.umich.edu`」の値に置き換え、タイトル「Manager」を追加し、`/tmp/modme.jpeg` ファイルの内容を `jpg` 写真として追加し、属性記述を完全に削除します。

このような変更は、以下のように `ldapmodify` の古い入力規則を使用して実行することもできます。

```
cn=Modify Me, o=University of Michigan, c=US
mail=modme@terminator.rs.itd.umich.edu
+title=Manager
+jpegPhoto=/tmp/modme.jpeg
-description
```

コマンドは次のようになります。

```
ldapmodify -b -r -f /tmp/entrymods
```

`/tmp/newentry` ファイルが存在すると仮定すると、次のような内容になります。

```
dn: cn=Barbara Jensen, o=University of Michigan, c=US
objectClass: person
cn: Barbara Jensen
cn: B Jensen
sn: Jensen
title: Manager
mail: bjensen@terminator.rs.itd.umich.edu
uid: bjensen
```

この場合、`/tmp/newentry` ファイルからの値を使用して、コマンド「`ldapadd -f /tmp/entrymods`」は「`B Jensen`」に新しいエントリを追加します。

`/tmp/newentry` ファイルが存在すると仮定すると、次のような内容になります。

```
dn: cn=Barbara Jensen, o=University of Michigan, c=US
changetype: delete
```

この場合、コマンド「`ldapmodify -f /tmp/entrymods`」は「`B Jensen`」のエントリを削除します。

## ldapdelete

`ldapdelete` ユーティリティは、指定したインデックスを削除します。LDAP サーバとの接続を開始し、バインドしてから削除します。構文は次のとおりです。

```
ldapdelete [-n] [-v] [-c] [-r] [-l] [-C] [-M] [-d debuglevel] [-e key
filename] [-f file] [-D binddn] [[-W] | [-w passwd]] [-h ldaphost] [-p
ldapport] [-Z[Z]] [dn]...
```

**注:** NetWare サーバでは、このユーティリティは、`ldelete` と呼ばれます。

`dn` パラメータは、削除するエントリの識別名のリストです。

これは、`-f` オプションと次のように通信します。

- ◆ コマンドラインに `-f` オプションがなく、コマンドラインで `dn` が指定されている場合、ユーティリティにより指定したエントリが削除されます。
- ◆ コマンドラインに `dn` と `-f` の両方がある場合、ユーティリティは `dn` のファイルを読み込んで削除して、コマンドラインの `dn` は無視します。
- ◆ コマンドラインに `dn` と `-f` オプションがない場合、ユーティリティは `stdin` から `dn` を読み込みます。

ヒント: ldap ユーティリティからの出力は stdout に送られます。出力が表示される前にユーティリティが存在する場合、出力はファイルにリダイレクトされます。例: ldapdelete [options] > out.txt

オプション	説明
-c	連続操作モード。エラーが通知されても、ldapdelete は削除動作を継続します。デフォルトでは、エラーが通知されると終了します。
-f file	ファイルから行を読み取り、1 行ごとに LDAP 検索を実行します。この場合、コマンドラインに指定されたフィルタは、%s が最初に出現した個所がファイルに指定された行で置換されるというパターンとして機能します。
-r	再起的に削除します。

注: 共通オプションについての詳細は、287 ページの「すべての LDAP ツールの共通オプション」を参照してください。

## 例

ldapdelete のコマンド「cn=Delete Me, o=University of Michigan, c=US」では、「University of Michigan」組織のエントリの真下にある commonName「Delete Me」で指定されたエントリを削除します。この場合、削除が許可されるためには binddn および passwd を指定する必要があります (-D オプションおよび -w オプションを参照)。

## ldapmodify

ldapmodify ユーティリティを使用すると、既存エントリの属性を変更したり、新規エントリを追加することができます。構文は次のとおりです。

```
ldapmodify [-a] [-c] [-C] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W] | [-w passwd]] [-h ldaphost] [-p ldap-port] [-P version] [-Z[Z]] [-f file]
```

注: NetWare サーバでは、このユーティリティは lmodify と呼ばれます。

-f オプションを指定すると、ldapmodify により変更がファイルから読み出されます。-f オプションを指定しない場合、変更は stdin から読み出されます。

ヒント: ldap ユーティリティからの出力は stdout に送られます。出力が表示される前にユーティリティが存在する場合、出力はファイルにリダイレクトされます。例: ldapmodify [options] > out.txt

オプション	説明
-a	新しいエントリを追加します。ldapmodify のデフォルトは、既存エントリの変更です。ldapadd として呼び出されると、このフラグは常にオンになります。
-r	デフォルトでは既存の値を置換します。
-c	連続操作モード。エラーが通知されても、ldapmodify は変更動作を継続します。デフォルトでは、エラーが通知されると終了します。
-f file	エントリ情報を標準入力ではなく LDIF ファイルから読み出します。レコードの最大長は 4096 行です。
-F	replica: で始まる入力行の内容に関わりなく、すべての変更を無条件で適用します (デフォルトでは、replica: 行は使用中の LDAP サーバホストおよびポートと比較され、repllog レコードを実際に適用するかどうか決定します)。

注: 共通オプションについての詳細は、287 ページの「すべての LDAP ツールの共通オプション」を参照してください。

## ldapmodrdn

ldapmodrdn を使用すると、エントリの相対識別名を変更できます。また、エントリを新しいコンテナに移動することもできます。構文は次のとおりです。

```
ldapmodrdn [-r] [-n] [-v] [-c] [-C] [-l] [-M] [-s newsuperior] [-d debuglevel]
[-e key filename] [-D binddn] [[-W] | [-w passwd]] [-h ldaphost] [-p ldapport]
[-Z[Z]] [-f file] [dn newrdn]
```

**注:** NetWare サーバでは、このユーティリティは <newrdn> と呼ばれます。

ldap ユーティリティからの出力は stdout に送られます。出力が表示される前にユーティリティが存在する場合、出力はファイルにリダイレクトされます。例: ldapmodrdn [options] > out.txt

オプション	説明
-c	連続操作モード。エラーが通知されても、ldapmodify は変更動作を続けます。デフォルトでは、エラーが通知されると終了します。
-f file	エントリ変更情報を標準入力やコマンドラインではなくファイルから読み出します。古い RDN と新しい RDN の間に空白行がないか確認します。空白行がある場合、-f オプションは失敗します。
-r	エントリから旧 RDN 値を削除します。デフォルトでは、旧値は保持されます。
-s newsuperior	エントリの移動先のコンテナの識別名を指定します。

**注:** 共通オプションについての詳細は、287 ページの「すべての LDAP ツールの共通オプション」を参照してください。

### 例

/tmp/entrymods ファイルが存在すると仮定すると、次のような内容になります。

```
cn=Modify Me, o=University of Michigan, c=US
```

```
cn=The New Me
```

## ldapsearch

ldapsearch ユーティリティは、指定された属性とオブジェクトクラスのディレクトリを検索します。構文は次のとおりです。

```
ldapsearch [-n] [-u] [-v] [-t] [-A] [-T] [-C] [-V] [-M] [-P] [-L] [-d
debuglevel] [-e key filename] [-f file] [-D binddn] [[-W] | [-w bindpasswd]]
[-h ldaphost] [-p ldapport] [-b searchbase] [-s scope] [-a deref] [-l time
limit] [-z size limit] [-Z[Z]] filter [attrs...]
```

**注:** NetWare サーバでは、このユーティリティは lsearch と呼ばれます。

ldapsearch ツールは LDAP サーバとの接続を開始し、バインドを行い、フィルタを使用して検索を実行します。フィルタは、RFC 2254 (<http://www.ietf.org/rfc/rfc2254.txt>) で定義された LDAP フィルタの文字列表現に準拠している必要があります。

ldapsearch が 1 つ以上のエントリを検出すると、attrs で指定された属性が取り込まれ、エントリと値が標準出力に書き込まれます。属性がリストされない場合、すべての属性が戻ります。

**ヒント:** ldap ユーティリティからの出力は stdout に送られます。出力が表示される前にユーティリティが存在する場合、出力はファイルにリダイレクトされます。例: ldapsearch [options] filter [attribute list] > out.txt

オプション	説明
-a <i>deref</i>	別名の逆参照の処理方法を指定します。次の値を使用します。 <ul style="list-style-type: none"> <li>◆ Never: ベースオブジェクトの検索時に、別名の逆参照は行われません。</li> <li>◆ Always: ベースオブジェクトの検索時に、別名の逆参照を常に行います。</li> <li>◆ Search: ベースオブジェクトの従属オブジェクトの検索時は別名の逆参照を行いますが、ベースオブジェクトの検索時には行いません。</li> <li>◆ Find: ベースオブジェクトの検索時は別名の逆参照を行いますが、ベースオブジェクトの従属オブジェクトの検索時には行いません。</li> </ul>
-A	値ではなく、属性のみ取り込まれます。エントリに属性が存在するかどうかを確認して、具体的な属性値を知る必要がない場合に便利です。
-CC	次の参照を有効にします。(同じバインド DN とパスワードで認証されたバインド)
-b <i>searchbase</i>	<i>searchbase</i> を検索の開始ポイントとして使用します。
-L	エントリを LDIF 形式で出力します。
-LL	エントリを LDIF 形式で出力します。コメントは出力されません。
-LLL	エントリを LDIF 形式で出力します。コメントおよびバージョンは出力されません。
-s <i>scope</i>	検索の範囲を指定します。範囲として、ベースオブジェクトを示す「base」、1 レベルを示す「one」、またはサブツリー検索を示す「sub」を指定します。デフォルトは「sub」です。
-S <i>attribute</i>	戻されたエントリを属性に基づいてソートします。デフォルトでは、戻ったエントリのソートを行いません。属性が長さ 0 の文字列( "") の場合、エントリはその識別名のコンポーネントによりソートされます。詳細については、 <i>ldap_sort</i> を参照してください。通常は、 <i>ldapsearch</i> はエントリを受け取った順に出力します。-S オプションを指定するとデフォルトが無効になり、すべてのエントリが取り込まれ、ソートされてから出力されます。
-t	検索されたバイナリ値が一時ファイルに書き込まれます。これは、jpeg の写真やオーディオなど ASCII 以外の値を扱うときに便利です。
-tt	すべての値が一時ファイルに書き込まれます。
-T <i>path</i>	ファイルをパス ( デフォルト : "/tmp" ) で指定されたディレクトリに書き出します。
-u	識別名 (DN) をユーザにわかりやすい形式で出力します。
-V	ファイルの URL プリフィックスです。
-V <i>prefix</i>	ファイルの URL プリフィックスを指定します ( デフォルト : "file://tmp/" ) 。
-z <i>sizelimit</i>	検索が終了するまで最大 <i>sizelimit</i> エントリだけ待機します。

注: 共通オプションについての詳細は、287 ページの「すべての LDAP ツールの共通オプション」を参照してください。



## 例

次のコマンドを実行します。

```
ldapsearch "cn=mark smith" cn telephoneNumber
```

**commonName** 「mark smith」のエントリのサブツリー検索 (デフォルトの検索ベースを使用) を実行します。**commonName** の値および **telephoneNumber** の値が取得され、標準出力に表示されます。2つのエントリが検出された場合、次のように入力されます。

```
cn=Mark D Smith, ou="College of Literature, Science, and the Arts",  
ou=Students, ou=People, o=University of Michigan, c=US
```

```
cn=Mark Smith
```

```
cn=Mark David Smith
```

```
cn=Mark D Smith 1
```

```
cn=Mark D Smith
```

```
telephoneNumber=+1 313,930-9489
```

```
cn=Mark C Smith, ou=Information Technology Division, ou=Faculty and Staff,  
ou=People,o=University of Michigan, c=US
```

```
cn=Mark Smith
```

```
cn=Mark C Smith 1
```

```
cn=Mark C Smith
```

```
telephoneNumber=+1 313 764-2277
```

コマンド :

```
ldapsearch -u -t "uid=mcs" jpegPhoto audio
```

デフォルトの検索ベースを使用して、ユーザ ID 「mcs」のエントリのサブツリー検索を実行します。エントリの DN は、DN 自体を含む行の後にわかりやすい形式で出力されます。また、**Jpeg** の写真の値およびオーディオの値が取得され、一時ファイルに書き込まれます。要求された属性がそれぞれ 1 つの値をもつエントリが 1 つ検出された場合、次のように入力されます。

```
cn=Mark C Smith, ou=Information Technology Division, ou=Faculty and Staff,  
ou=People, o=University of Michigan, c=US
```

```
Mark C Smith, Information Technology Division, Faculty and Staff, People,  
University of Michigan, US
```

```
audio=/tmp/ldapsearch-audio-a19924
```

```
jpegPhoto=/tmp/ldapsearch-jpegPhoto-a19924
```

次のコマンドは、**organizationName** が 「university」 で始まるすべての組織の **c=US** レベルで、1 レベルの検索を実行します。

```
ldapsearch -L -s one -b "c=US" "o=university*" o description
```

検索結果は LDIF 形式で表示されます。**organizationName** の値および記述属性の値が取得され、標準出力に書き込まれます。出力結果の例は次のようになります。

```
dn: o=University of Alaska Fairbanks, c=US
```

```
o: University of Alaska Fairbanks
```

```
description: Preparing Alaska for a brave new yesterday.
```

```

description: leaf node only
dn: o=University of Colorado at Boulder, c=US
o: University of Colorado at Boulder
description: No personnel information
description: Institution of education and research
dn: o=University of Colorado at Denver, c=US
o: University of Colorado at D

```

## ndsindex

ndsindex を使用すると、インデックスの作成、一覧表示、一時停止、再開、削除を実行できます。構文は次のとおりです。

```
ndsindex list [-h <hostname>] [-p <port>] -D <bind DN> -W[[-w <password>]] [-l limit] -s <eDirectory Server DN> [-Z[Z]] [<indexName1>, <indexName2>.....]
```

```
ndsindex add [-h <hostname>] [-p <port>] -D <bind DN> -W[[-w <password>]] [-l limit] -s <eDirectory Server DN> [-Z[Z]] <indexDefinintion1> [<indexDefinintion2>.....]
```

```
ndsindex delete [-h <hostname>] [-p <port>] -D <bind DN> -W[[-w <password>]] [-l limit] -s <eDirectory Server DN> [-Z[Z]] <indexName1> [<indexName2>.....]
```

```
ndsindex resume [-h <hostname>] [-p <port>] -D <bind DN> -W[[-w <password>]] [-l limit] -s <eDirectory Server DN> [-Z[Z]] <indexName1> [<indexName2>.....]
```

```
ndsindex suspend [-h <hostname>] [-p <port>] -D <bind DN> -W[[-w <password>]] [-l limit] -s <eDirectory Server DN> [-Z[Z]] <indexName1> [<indexName2>.....]
```

注: NetWare サーバでは、このユーティリティは nindex と呼ばれます。

オプション	説明
list	指定したインデックスを表示します。インデックスが指定されていない場合、ndsindex はサーバ上のすべての既存のインデックスを表示します。
add	新しいインデックスを作成します。
delete	指定したインデックスを削除します。
resume	指定したインデックスをオフラインの状態から再開します。
suspend	指定したインデックスを一時停止してオフラインの状態にします。
-s <i>eDirectory Server DN</i>	eDirectory サーバの DN を指定します。

注: 共通オプションについての詳細は、287 ページの「すべての LDAP ツールの共通オプション」を参照してください。

### 例

サーバ MyHost 上のインデックスを表示するには、次のコマンドを入力します。

```
ndsindex list -h MyHost -D cn=admin, o=mycompany -w password -s cn=MyHost, o=novell
```

電子メールの属性に **MyIndex** という名前の下位文字列インデックスを作成するには、次のコマンドを入力します。

```
ndsindex add -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell "MyIndex;email address;substring"
```

市町村の属性に **MyIndex** という名前の値インデックスを作成するには、次のコマンドを入力します。

```
ndsindex add -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell "MyIndex;city;value"
```

自宅電話番号の属性に **MyIndex** という名前の存在インデックスを作成するには、次のコマンドを入力します。

```
ndsindex add -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell "MyIndex;homehone;presence"
```

**MyIndex** という名前のインデックスを削除するには、次のコマンドを入力します。

```
ndsindex delete -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell MyIndex
```

**MyIndex** という名前のインデックスを一時停止するには、次のコマンドを入力します。

```
ndsindex suspend -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell MyIndex
```

**MyIndex** という名前のインデックスを再開するには、次のコマンドを入力します。

```
ndsindex resume -h myhost -D cn=admin, o=mycompany -w password -s cn=myhost, o=novell MyIndex
```

## 拡張可能一致検索フィルタ

[RFC 2251 \(http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2251.html\)](http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2251.html) で定義される LDAP 3 のコアプロトコルを指定するには、LDAP サーバが拡張可能一致検索フィルタ機能を認識できなくてはなりません。拡張可能一致検索では、LDAP クライアントは検索フィルタに次の項目を指定することができます。

- ◆ オプションの属性名
- ◆ オプションの一致ルール
- ◆ dn 属性がエントリの一部として考慮されるべきかを示すフラグ
- ◆ 一致検索に使用される値

拡張可能一致検索フィルタの文字列を次に示します。

```
extensible = attr [":dn"] [": matchingrule"] ":@" value /  
[:dn"] ":@"matchingrule ":@" value
```

次の表は、拡張可能検索フィルタパラメータを表示したものです。

パラメータ	説明
attr	適合する属性を指定します。
[:dn"]	一致ルールが比較一致に含まれていることを示します。

パラメータ	説明
[":" matchingrule]	使用する一致ルールを指定します。
":="	一致ルールを指定しないと、完全一致とみなされます。
value	比較値

extensibleMatch は、LDAP 3 から導入された新しいフィルタです。matchingRule フィールドがない場合は、属性フィールドが必ず必要です。完全一致検索はその属性に対して実行されます。attribute フィールドがなく、matchingRule が存在する場合、その matchingRule をサポートするエントリ内のすべての属性が matchValue と比較され、matchingRule によりアサーション値の構文が決定されます。

フィルタ項目は次のように評価されます。

- ◆ TRUE: エントリに 1 件以上の一致があることを表します。
- ◆ FALSE: エントリに一致する属性がないことを表します。
- ◆ matchingRule を認識できなかつたり、assertionValue を解析できない場合は「未定義」とされます。

matchingRule の他に type フィールドがある場合、matchingRule はその type で使用できるものでなくてはなりません。使用できない場合、フィルタ項目は定義されません。検索フィルタに :dn が指定されている場合、エントリの識別名に含まれるすべての属性に対して一致検索が適用されます。また、フィルタ項目の評価が TRUE の識別名が 1 つ以上属性を持っている場合も、評価は TRUE になります。dnAttributes フィールドを使用すると、単語の一致検索などで、1 つのルールをエントリに適用し、別のルールをエントリと dn 属性に適用するというように、一般的な一致ルールを複数設定する必要がなくなります。

拡張可能一致検索フィルタにより、LDAP クライアントでは次の 2 つのことが可能になります。

- ◆ 同じタイプのデータに対し、複数の一致ルールをサポートできます。
- ◆ 検索条件に DN 要素を含めることができます。

DN 指定により、DN の特定要素の一致検索を実行できます。

Novell eDirectory 8.7.3 では、DN の拡張可能な一致検索フィルタをサポートしています。拡張可能一致検索フィルタのもう一つの要素である一致ルールは未定義とみなされ、無視されます。DN 一致検索を使用すると、LDAP クライアントで eDirectory ツリーからオブジェクトを簡単に検索できます。たとえば、

```
(&(ou:dn:=sales)(objectclass=user))
```

のような複雑な LDAP 検索フィルタにより、セールスコンテナの下のセールスファンクションにあるすべてのユーザオブジェクトをリストすることができます。

## 使用例

次は、eDirectory 8.7.3 でサポートされている拡張可能一致検索フィルタの文字列の例です。

```
(o:dn:=Ace Industry)
```

これは :dn の使用例です。エントリの識別名の属性は、一致を評価するとき、エントリの一部とみなされます。これは、完全一致であることを意味します。

```
(:dn:2.4.8.10:=Dino)
```

これはエントリの属性に適用するフィルタの例です。一致ルールが 2.4.8.10 の DN の属性も考慮されます。

次は、eDirectory 8.7.3 でサポートされていない拡張可能一致検索フィルタの文字列の例です。

```
(cn:1.2.3.4.5:=John Smith)
```

この例は、属性タイプ **cn** と値 **John Smith** を指定するフィルタを表しています。一致ルール **oid 1.2.3.4.5** により、ディレクトリサーバにより一致検索が実行されます。

```
(sn:dn:2.4.6.8.10:=Barbara Jones)
```

これは **dn** の使用例です。一致ルール **2.4.6.8.10** を比較に使用して一致を評価するとき、エントリの属性の識別名はエントリの一部とみなされることを表しています。



# 12 LDAP Services for Novell eDirectory の環境設定

eDirectory™ インストールプログラムにより、LDAP Services for Novell® eDirectory が自動的にインストールされます。eDirectory のインストールについては、『*Novell eDirectory 8.7.3 インストールガイド*』を参照してください。

このセクションでは次について説明します。

- ◆ 299 ページの「LDAP Services for eDirectory をロードおよびアンロードする」
- ◆ 300 ページの「LDAP サーバがロードされているか確認する」
- ◆ 301 ページの「LDAP サーバが実行されているか確認する」
- ◆ 304 ページの「LDAP オブジェクトを環境設定する」
- ◆ 309 ページの「LDAP サーバをリフレッシュする」
- ◆ 310 ページの「認証とセキュリティ」
- ◆ 317 ページの「LDAP サーバを使ってディレクトリを検索する」
- ◆ 324 ページの「上方参照を設定する」
- ◆ 329 ページの「持続的検索 :eDirectory イベントの設定」
- ◆ 331 ページの「LDAP サーバの情報を取得する」

LDAP ツールの詳細については、[LDAP Tools \(http://developer.novell.com/ndk/doc/cldap/index.html?ldaplibc/data/a6eup29.html\)](http://developer.novell.com/ndk/doc/cldap/index.html?ldaplibc/data/a6eup29.html) を参照してください。

## LDAP Services for eDirectory をロードおよびアンロードする

LDAP Services for eDirectory をロードするには、次のコマンドを入力します。

サーバ	コマンド
NetWare の場合®	コンソールプロンプトで、次のコマンドを入力します。  <code>load nldap.nlm</code>
Windows	[DHOST (NDSCONS)] 画面で nldap.dlm を選択し、[開始] をクリックします。
Linux、Solaris、AIX、または HP-UX	Linux、Solaris、AIX、または HP-UX プロンプトで、次のコマンドを入力します。  <code>/usr/sbin/nldap -l</code>

LDAP Services for eDirectory をアンロードするには、次のコマンドを入力します。

サーバ	コマンド
NetWare	<p>コンソールプロンプトで、次のコマンドを入力します。</p> <pre>unload nldap.nlm</pre>
Windows	<p>[DHOST (NDSCONS)] 画面で nldap.dlm を選択し、[停止] をクリックします。</p>
Linux、Solaris、AIX、HP-UX	<p>LDAP をアンロードするには、[DHOST リモート管理] ページで、[LDAP v3 for Novell eDirectory 8.7.3] アクションアイコンをクリックして停止します。</p> <p>または</p> <p>Linux、Solaris、AIX、または HP-UX プロンプトで、次のコマンドを入力します。</p> <pre>/usr/sbin/nldap -u</pre>

## LDAP サーバがロードされているか確認する

LDAP オブジェクトの設定を行う前に、LDAP サーバがロードされ、動作していることを確認します。この画面では、LDAP サーバがロードされているか確認する方法を説明します。サーバが動作しているか確認するには、[301 ページの「LDAP サーバが実行されているか確認する」](#)を参照してください。

### NetWare の場合

NetWare サーバに nldap.nlm がロードされているか確認するには、サーバコンソールに次のいずれかを入力します。

- ◆ **ldap display activity**


nldap.nlm がロードされていない場合は、サーバにより「不明なコマンド」というメッセージが表示されます。

NetWare 6x では、メッセージはコンソール画面ではなくログ画面に表示されます。

- ◆ **ldap display config**

- ◆ **modules nldap.nlm**

Novell iManager を使用することもできます。

- 1 [役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [サービスマネージャ] の順にクリックします。
- 3 接続、サーバ、DNS 名、または IP アドレスを選択し、[OK] をクリックします。
- 4 パスワードを入力し、[OK] をクリックします。
- 5 [LDAP Agent for Novell eDirectory 8.7.3] をクリックします。  
モジュール情報セクションの[ファイル名]フィールドに nldap.nlm と表示されます。


### Windows NT/2000 の場合

- 1 Windows サーバで、ndscons.exe を開きます。  
[スタート] > [設定] > [コントロールパネル] > [Novell eDirectory Services] の順にクリックします。



- 2 [サービス] タブで、`nldap.dlm` までスクロールし、[ステータス] カラムを表示します。  
カラムに「稼働中」と表示されます。

Novell iManager を使用することもできます。

- 1 [役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [サービスマネージャ] の順にクリックします。
- 3 接続、サーバ、DNS 名、または IP アドレスを選択し、[OK] をクリックします。
- 4 パスワードを入力し、[OK] をクリックします。
- 5 [LDAP Agent for Novell eDirectory 8.7.3] をクリックします。  
モジュール情報セクションの[ファイル名]フィールドに `nldap.nlm` と表示されます。

### UNIX にロードした場合

`libnldap.so` または `libnldap.sl` を指定します。この名前は、単にバージョン情報が追加された長いファイル名のシンボリックリンクであることもあります。

また、`libnldap.so` または `libnldap.sl` ファイルは、各 UNIX プラットフォーム用の異なるバイナリファイルです。

LDAP サーバがロードされているかどうかを確認するのに、`ndslog` ファイルまたは `ndstrace` を使用することもできます。

## LDAP サーバが実行されているか確認する

LDAP サーバをロードした後で、それが実行されているか確認します。その後、デバイスが監視しているか確認します。

- ◆ [301 ページの「シナリオ」](#)
- ◆ [302 ページの「LDAP サーバが実行されているか確認する」](#)
- ◆ [304 ページの「デバイスがリッスンしていることを確認する」](#)

## シナリオ

通常、LDAP サーバはロードされるとすぐに実行されます。ただし、次の 2 つのシナリオでは、サーバが正しく実行されないことがあります。

**シナリオ : サーバがゾンビ状態にある** LDAP サーバは NetWare か DHost ロダが外部依存関係を解決できる限り、ロードを続けます。ただし、LDAP サーバは有効な設定を 2 つの設定オブジェクト (LDAP サーバと LDAP グループオブジェクト) から取得するまで、正しく動作しません。

LDAP サーバが、「ロードはされても、実行されない (ゾンビ) 状態」の場合、LDAP サーバは定期的に設定オブジェクトを探そうとします。オブジェクトの設定が失敗したり破損した場合、LDAP サーバはサーバ (`nldap.nlm`、`nldap.dlm`、`libnldap.so`、または `libnldap.sl`) をアンロードまたは終了するまで、ゾンビ状態のままになります。

ローダは、LDAP サーバがロードされているのに、`nldap.nlm` (または `nldap.dlm`、`libnldap.so`、`libnldap.sl`) により LDAP ポート (389、636) が開かれていないことを示します。また、LDAP クライアントの要求はどれも実行されていません。

定期的な試行の記録と、サーバが稼働状態にならない原因を示す DTrace メッセージが表示されます。

シナリオ : サービス拒否 Digital Airlines 社のサーバは現在、長時間 (20 分以上) かかる検索を処理しています。この処理では、膨大なデータの中から検索が行われています。

この検索の実行中に、あるユーザが次のいずれかを実行します。

- ◆ 環境設定パラメータを変更し、設定オブジェクトを更新する。
- ◆ [Refresh Server Now(サーバをリフレッシュ)] をクリックする。
- ◆ LDAP サーバ (nldap.nlm、nldap.dlm、libnldap.so、libnldap.sl のいずれか) をアンロードする。
- ◆ サーバ全体を終了しようとする。

LDAP サーバは、現在の処理が完了してから更新を適用します。この更新が完了するまで、新しい操作は実行されません。この遅れにより、サーバの検索が完了し更新が適用されるまでの間、サーバが新しいリクエストに応答しなくなったか、またはアンロードの実行中にサーバが停止したかのように感じられることがあります。

検索要求が長く、多くの一致項目がある場合、LDAP サーバをアンロードしようとする と検索は中止され、次の一致項目がクライアントに返される時にアンロードが実行されます。ただし、検索要求の結果、20 分間に 1 件以下しか一致する項目が見つからなかった場合、LDAP サーバは実行中の NDS<sup>®</sup> または eDirectory 要求を中止できません。

リフレッシュまたは更新の場合、クライアントに返される多くの一致レコードがあっても検索は中止されません。

## LDAP サーバが実行されているか確認する

LDAP サービスが実行されているか確認するには、Novell インポート / エクスポート変換ユーティリティ (ICE) を使用します。ワークステーションで、コマンドラインから `ice.exe` を実行するか、または Novell iManager か ConsoleOne<sup>®</sup> を使用します。

### コマンドラインから実行する方法

- 1 `ice.exe` を含むディレクトリに移動します (例 `:c:\novell\consoleone\1.2\bin`)。
- 2 `ice.exe` を実行します。

`rootDSE`を検索します。ソースハンドラとエクスポートハンドラを特定するパラメータを含めます。たとえば、次のように入力します。

```
ice -S LDAP -s 10.128.45.0 -p 389 -c base -a vendorname -D LDIF -f testoutput
```

パラメータと値	説明
-S LDAP	LDAP をソースハンドラとして指定します。
-s 10.128.45.0	サーバの DNS 名または IP アドレスを指定します。
-p 389	LDAP ソースハンドラパラメータが識別する LDAP サーバのポート番号を指定します。デフォルトポートは 389 です。389 がインストール時に指定したポートではない場合、クリアテキストのポート番号を指定します。
-c base	検索要求の範囲として、ベースオブジェクトのエントリのみを指定します。

パラメータと値	説明
-a vendorname	検索対象に vendorname 属性を指定します。
-D LDIF	LDIF をターゲットハンドラとして指定します。
-f testoutput	LDIF レコードの書き込み先になるファイルの名前を指定します。

これは、出力を testoutput ファイルに送信したときの例です。

ICE の使用方法については、131 ページの「Novell インポート / エクスポート変換ユーティリティ」を参照してください。LDAP ソースハンドラに固有の情報については、139 ページの「LDAP ソースハンドラのオプション」を参照してください。LDIF ターゲットハンドラに固有の情報については、138 ページの「LDIF ターゲットハンドラのオプション」を参照してください。

### 3 ICE コマンドの結果が表示されます。

```

C:\Novell\CONSOL~1\1.2\bin>type testoutput
#This LDIF file was generated by Novell's ICE and the LDIF destination handler.
version: 1

dn:
changetype: add
vendorname: Novell, Inc.

```

例 (手順 2 と 3) では rootDES エントリからの出力をベンダ名属性のみに制限しています。この例では、情報が Novell eDirectory サーバから読み込まれたため、ベンダ情報に Novell, Inc. と表示されます。

### Novell iManager を使用する方法

Novell iManager で LDAP サーバが動作していることを確認するには、133 ページの「データをファイルへエクスポートする」の手順に従います。

IP アドレスとポート番号を入力して接続が確立された場合は、サーバは機能しています。その他の場合は、エラーメッセージが表示されます。ログファイルまたはエクスポートファイルをダウンロード (表示) してください。

### ConsoleOne を使用する方法

ConsoleOne を使用して LDAP サーバが動作していることを確認するには、147 ページの「LDIF エクスポートを実行する」を参照してください。

[ターゲット LDIF ファイルの選択] フィールドにパスとファイル名を指定します (例 :c:\ldap\textoutput.txt)。ファイル名のみ入力した場合、ConsoleOne の LDAP スナップインは、ファイルをデフォルトディレクトリ (通常、c:\novell\consoleone\1.2\bin) に書き出します。

## デバイスがリッスンしていることを確認する

デバイスがポート 389 でリッスンしていることを確認します。

### NetWare の場合：

- 1 サーバコンソールで次のように入力します。

```
tcpcon
```

- 2 [プロトコル情報] > [TCP] > [TCP 接続] の順に選択します。

- 3 [ポート] カラムで 389 を選択します。

[状態] カラムに [リッスン] と表示された場合、デバイスはそのポートでリッスンしています。

デバイスがリッスンしていない場合、そのポートは存在しません。

### Windows 2000/NT および UNIX の場合

- 1 コマンドラインで次のように入力します。

```
netstat -a
```

- 2 ローカルアドレスが *servername:389* で、状態が「リッスン中」の行を探します。

次のいずれかの状況が発生した場合は、Novell iMonitor を実行します。

- ◆ ICE ユーティリティから情報を取得できない
- ◆ LDAP サーバが LDAP 要求を処理しているかどうか確認できない

Novell iMonitor の詳細については、[177 ページの「環境設定ファイル」](#)および [184 ページの「トレースを環境設定する」](#)を参照してください。

LDAP 要求の詳細については、『*Novell eDirectory 8.7.3 インストールガイド*』の「[LDAP を介した eDirectory との通信](#)」を参照してください。

## LDAP オブジェクトを環境設定する

eDirectory のインストール時に、LDAP サーバオブジェクトと LDAP グループオブジェクトが作成されます。LDAP サービスのデフォルト設定は、これらの 2 つのオブジェクト上のディレクトリにあります。ConsoleOne LDAP スナップインか、Novell iManager の LDAP 管理タスクを使用して、デフォルト設定を変更できます。

LDAP サーバオブジェクトとは、サーバ固有の設定データのことです。

LDAP グループオブジェクトには、複数の LDAP サーバ間で共有できる便利な設定情報が含まれています。このオブジェクトは、共通の設定データと LDAP サーバグループを提供します。サーバは共通データを持っています。

複数の LDAP サーバオブジェクトを、1 つの LDAP グループオブジェクトと関連させることができます。関連するすべての LDAP サーバは、サーバ固有の設定を LDAP サーバオブジェクトから取得しますが、共通する情報や共有情報は LDAP グループオブジェクトから取得します。

デフォルトで、eDirectory インストールプログラムは `nldap.nlm` または `nldap.dlm` に LDAP グループオブジェクトと LDAP サーバオブジェクトを 1 つずつインストールします。その後、複数の LDAP サーバオブジェクトを、1 つの LDAP グループオブジェクトに関連付けることができます。

**重要:** 新しいバージョンの LDAP サーバオブジェクトを古いバージョンの LDAP グループオブジェクトに関連付けることも可能ですが、異なるバージョン間での関連付けはお勧めできません。たとえば、eDirectory 8.5 の LDAP グループオブジェクトと eDirectory 8.6 の LDAP サーバオブジェクトとの関連付けは避けるようにしてください。

LDAP グループオブジェクトが保持する共通情報の量は制限されています。属性に含まれるデータはほとんど共通しているため、LDAP は多くの属性を読み込む必要がありません。多くの LDAP サーバは同じデータを使用する必要があります。共通の、または共有グループオブジェクトがない場合は、各 LDAP サーバにそのデータを複製する必要があります。

LDAP サーバオブジェクトでは、サーバ固有の設定オプションおよびデータが、LDAP グループオブジェクトの場合よりも多く許可されています。

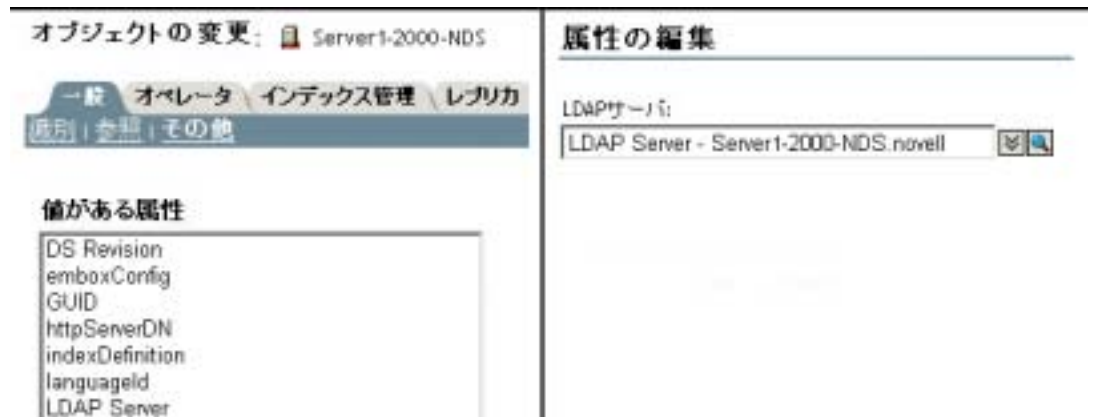
どちらのオブジェクトも DN 構文属性を持ち、この属性を使用して互いを指定します。

LDAP サーバがその設定データを検索できるようにするには、関連付けを追加する必要があります。関連付けは、通常の eDirectory 設定データを保持する NCP™ サーバを通じて行われます。この関連付けは、eDirectory インストールプログラムにより自動で行われます。

各 eDirectory サーバは、NCP サーバオブジェクトを持っています。次の図の「Lundi」というサーバには、このオブジェクトが iManager 上と同じように表示されています。



このオブジェクトは、特定のホスト eDirectory サーバの LDAP サーバオブジェクトを指す LDAP サーバ属性を持っています。次の図は、この属性を示しています。



通常、LDAP サーバオブジェクト、LDAP グループオブジェクト、NCP サーバオブジェクトは同じコンテナ内にあります。eDirectory のインストールで、サーバおよび管理者コンテキストを指定するときに、このコンテナを指定します。

LDAP サーバオブジェクトを移動するときは、それを書き込み可能なレプリカ上に移動する必要があります。

## Linux、Solaris、AIX、HP-UX システム上で、LDAP サーバオブジェクトおよび LDAP グループオブジェクトを環境設定する

LDAP 環境設定ユーティリティは `ldapconfig` です。Linux、Solaris、AIX、HP-UX システムで LDAP サーバオブジェクトおよび LDAP グループオブジェクトの属性を変更、表示、リフレッシュするには、`ldapconfig` を使用します。

Linux、Solaris、AIX、HP-UX システム上で LDAP 属性値を表示するには、次の構文を使用します。

```
ldapconfig get [...] | set attribute-value-list [-t treename | -p  
hostname[:port]] [-w password] [-a user FDN] [-f]
```

```
ldapconfig [-t tree_name | -p host_name[:port]] [-w password] [-a user FDN]  
[-V] [-R] [-H] [-f] -v attribute,attribute2...
```

Linux、Solaris、AIX、HP-UX システム上で LDAP 属性を変更するには、次の構文を使用します。

```
ldapconfig [-t tree_name | -p host_name[:port]] [-w password] [-a admin_FDN]  
-s attribute=value,...
```

パラメータ	説明
<code>-t treename</code>	コンポーネントのインストール先となる eDirectory ツリーの名前。
<code>-p hostname</code>	ホストの名前です。DNS名またはIPアドレスを指定することもできます。
<code>-w</code>	管理権を持つユーザのパスワード。
<code>-a</code>	管理権を持つユーザの完全識別名。次に例を示します。 <code>cn=user.o=org1</code>
<code>get   -V</code>	すべての LDAP サーバと LDAP グループの属性を表示します。
<code>get   -v attribute list</code>	属性リストにある属性の現在の値を表示します。
<code>set   -s attribute-value pairs</code>	属性を指定した値で設定します。
<code>-v</code>	LDAP 属性値を表示します。
<code>-s</code>	インストールされたコンポーネントの属性値を設定します。
<code>-R</code>	LDAP サーバをリフレッシュします。
<code>-V</code>	現在の LDAP 環境設定を表示します。
<code>-H</code>	使用方法とヘルプを表示します。
<code>-f</code>	フィルタ済みレプリカ上での操作を許可します。

パラメータ	説明
<i>attribute</i>	設定可能な LDAP サーバ属性名またはグループ属性名。詳細については、 <a href="#">307 ページの「LDAP サーバオブジェクトの属性」</a> および <a href="#">309 ページの「LDAP グループオブジェクトの属性」</a> を参照してください。

## 例

属性リストの属性の値を表示するには、次のコマンドを入力します。

```
ldapconfig [-t tree_name | -p host_name[:port]]
  [-w password] [-a user_FDN] -v "Require TLS for simple binds with
  password","searchTimeLimit"
```

LDAP TCP ポート番号と検索サイズの制限を 1000 に設定するには、次のコマンドを入力します。

```
ldapconfig [-t tree_name | -p host_name[:port]]
  [-w password] [-a admin_FDN] -s "LDAP TCP
  Port=389","searchSizeLimit=1000"
```

## LDAP サーバオブジェクトの属性

Novell LDAP サーバプロパティを設定および管理するために、LDAP サーバオブジェクトを使用します。

次の表に、LDAP サーバ属性の説明を示します。

属性	説明
LDAP Server	eDirectory の LDAP サーバオブジェクトの完全識別名。
LDAP Host Server	LDAP サーバの実行場所となるホスト eDirectory サーバの完全識別名。
LDAP Group	eDirectory の中で、この LDAP サーバがメンバーとして属する LDAP グループオブジェクト。
LDAP Server Bind Limit	LDAP サーバに同時にバインドできるクライアントの数。0 を指定すると、無制限になります。
LDAP Server Idle Timeout	あるクライアントと LDAP サーバの間で、ここで指定した期間無活動状態が継続すると、このクライアントと LDAP サーバの接続が切断されます。0 を指定すると、無制限になります。
LDAP Enable TCP	この LDAP サーバに対して (TLS ではなく) TCP 接続が有効であるかどうかを示します。 値は 1(はい)または 0(いいえ)です。
LDAP Enable TLS	この LDAP サーバに対して TLS 接続が有効であるかどうかを示します。 値は 1(はい)または 0(いいえ)です。

属性	説明
LDAP TCP Port	(SSL でなく )TCP 接続で LDAP サーバがリスンするポート番号。  範囲は 0 ~ 65535 です。
LDAP TLS Port	TLS 接続で LDAP サーバがリスンするポート番号。  範囲は 0 ~ 65535 です。これは LDAP サーバに許可されている最大接続数です。
keyMaterialName	この LDAP サーバに関連付けられた、SSL LDAP 接続に使用する eDirectory の証明書オブジェクトの名前。
searchSizeLimit	LDAP サーバが検索要求への応答として LDAP クライアントに返すエントリの最大数。0 を指定すると、無制限になります。
searchTimeLimit	LDAP サーバによる LDAP 検索がタイムアウトになるまでの最大秒数。0 を指定すると、無制限になります。
filteredReplicaUsage	LDAP サーバが LDAP 検索のために、フィルタ処理されたレプリカを使用するかどうかを指定します。  値は 1( フィルタ済みレプリカを使用 )か 0( フィルタ済みレプリカ不使用 )です。
sslEnableMutualAuthentication	LDAP サーバにおいて、SSL ベースの相互認証 ( 証明書に基づくクライアント認証)を有効にするかどうかを指定します。
ldapTLSVerifyClientCertificate	LDAP による TLS 操作のクライアント認証の確認を有効または無効にします。
ldapNonStdAllUserAttrsMode	非標準のすべてのユーザとオペレーショナル属性を有効または無効にします。
ldapBindRestrictions	LDAP クライアント接続で LDAP のバインド制限を設定します。LDAP クライアントから匿名バインドを許可または禁止することができます。  値は 0 か 1 です。  0 を指定すると、クライアントからの匿名バインドが許可されます。1 を指定すると、クライアントの匿名バインドが制限されます。
ldapEnablePSearch	LDAP サーバで持続的検索機能を有効にするかどうかを指定します。  値は TRUE か FALSE です。
ldapMaximumPSearchOperations	同時に実行できる持続的検索操作の数を制限するための整数値です。0 を指定すると、検索操作は無制限になります。



属性	説明
ldapignorePSearchLimitsForEvents	<p>持続的検索要求によって最初の結果が返された後で、サイズと時間の制限を無視するかどうかを指定します。</p> <p>値は TRUE か FALSE です。</p> <p>この属性が FALSE に設定されている場合、すべての持続的検索操作は検索制限の制約を受けます。サイズと時間のいずれかの制限に達した場合、検索操作は失敗し、該当するエラーメッセージが返されます。</p>

## LDAP グループオブジェクトの属性

LDAP クライアントの Novell LDAP サーバに対するアクセス方法と、サーバ上の情報の使用方法を設定および管理するには、LDAP グループオブジェクトを使用します。

単純バインドに TLS が必要な場合は、[310 ページの「パスワードとの単純バインドに TLS を要求する」](#)を参照してください。この属性は、LDAP サーバが LDAP クライアントからパスワードをクリアテキストで送信することを許可するかどうかを指定します。値は 1(はい) または 1(いいえ) です。

デフォルトの参照および LDAP サーバによる LDAP 参照の処理方法を指定するには、[318 ページの「参照を使用する」](#)を参照してください。

## LDAP サーバをリフレッシュする

LDAP サーバの環境設定オプションや LDAP サーバの設定を変更した場合、変更を有効にするにはサーバをリフレッシュする必要があります。

ただし、LDAP 要求のサービスの実行中はサーバをリフレッシュできません。たとえば、eDirectory ツリーの処理に 15 分かかる場合には、この処理が完了するまでリフレッシュは実行されません。

同様に、LDAP サーバスレッドの実行中は、LDAP サーバを終了することはできません。

リフレッシュの実行が予定されている場合は、LDAP サーバはリフレッシュが実行されるまで新しい LDAP 要求の開始を遅らせます。

デフォルトでは、LDAP サーバは 30 分間隔で LDAP サーバオブジェクトと LDAP グループオブジェクトのタイムスタンプをチェックし、設定に変更がなかったか確認します。設定が変更されている場合、サーバはその変更を適用します。

設定のタイムスタンプが前回と変わらない場合には、リフレッシュは実行されません (強制的にリフレッシュを実行すると、サーバはタイムスタンプを無視して変更を適用します)。

LDAP サーバをリフレッシュするには、次のいずれかを実行します。

- ◆ Novell iManager を使用する。
  1. 役割およびタスクページで、[LDAP] > [LDAP の概要] > [LDAP サーバの表示] の順にクリックします。
  2. LDAP サーバをクリックし、[リフレッシュ] をクリックします。
- ◆ サーバが次のリフレッシュ間隔で再設定されるまで待つ。

- ◆ `nldap.nlm` をアンロードしてから再ロードする。

`nldap.nlm` をアンロードする前に、前提となる NLM™ プログラムをアンロードする必要はありません。

`nldap.nlm` がアンロードされると、従属する NLM プログラムが再ロードされます。

- ◆ コマンドラインで、リフレッシュ間隔を変更する。

このオプションは、WAN リンクが継続して実行されていない場合に便利です。必要に応じ、一時的にサーバのハートビート処理の長さを変更できます。

この変更は持続しません。`nldap.nlm` をロードするたびに、コマンドを再入力する必要があります。

サーバコンソールで次のように入力します。

**`ldap refresh [=] [date][time][interval]`**

- ◆ 日付変数の形式は、`mm:dd:yyyy` です。すべての日付フィールドに 0 と入力すると、現在の日付が使用されます。
- ◆ 時間変数の形式は、`hh:mm:ss` です。すべての時刻フィールドに 0 と入力すると、現在の時刻が使用されます。
- ◆ 間隔変数の形式は 0 または 1～2147483647 分の間です。0 と入力すると、デフォルトの 30 分が使用されます。

このコマンドは、`sys:\system` ディレクトリの `autoexec.ncf` ファイルに追加できます。`nldap.nlm` をロードした行の後に、このコマンドを配置します。

## 認証とセキュリティ

このセクションでは、次の情報について説明します。

- ◆ 310 ページの「パスワードとの単純バインドに TLS を要求する」
- ◆ 311 ページの「TLS を開始 / 停止する」
- ◆ 312 ページの「TLS のサーバを環境設定する」
- ◆ 313 ページの「TLS のクライアントを環境設定する」
- ◆ 313 ページの「ルート認証局をエクスポートする」
- ◆ 314 ページの「クライアント証明書で認証を受ける」
- ◆ 314 ページの「サードパーティプロバイダの証明書を使用する」
- ◆ 316 ページの「SASL を使用する」


### パスワードとの単純バインドに TLS を要求する

SSL(Secure Socket Layer) 3.1 は Netscape でリリースされました。IETF は TLS(Transport Layer Security) 1.0 を実装することにより、SSL 標準の所有権を持っています。

TLS を使用すると、接続をセッション層で暗号化することができます。TLS 接続のために、暗号化されたポートを使用する必要はありません。また、次の方法もあります。ポート 636 は暗黙的な TLS ポートであり、クライアントがセキュアポートに接続すると、LDAP サーバは自動的に TLS 接続を開始することができます。

クライアントは、まずクリアテキストポートに接続し、後で TLS を使用して暗号化された接続にアップグレードすることもできます。

パスワードとの単純バインドに TLS を要求するには、次を実行します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] > [LDAP グループの表示] の順にクリックします。
- 3 LDAP グループオブジェクトをクリックしてから、[全般] タブの [情報] をクリックします。
- 4 [パスワードとの単純バインドに TLS を必要とする] チェックボックスをオンにします。



- 5 [適用] をクリックし、[OK] をクリックします。

## TLS を開始 / 停止する

LDAP 拡張オペレーション STARTTLS により、クリア接続から暗号化された接続にアップグレードすることができます。このアップグレードは、eDirectory8.7 の新機能です。

暗号化された接続を使用すると、パケット全体が暗号化されます。このため、ネットワーク経由で送信されたデータが第三者によって診断されることはありません。

**シナリオ : STARTTLS を使用する** — ポート 389 にクリア接続し、匿名検索を行います。ただし、セキュリティ保護されたデータを扱う場合には TLS セッションに切り換えます。拡張オペレーション STARTTLS を実行し、クリア接続から暗号化された接続にアップグレードします。これでデータの安全が確保されます。

暗号化されたセッションをクリア接続に切り替えるには、TLS を停止します。クリア接続では、クライアントが送受信するデータは暗号化および解読されないで、負荷は少なくなります。そのため、クリア接続の使用時の方が、データの通信速度が速くなります。この時点で、接続は匿名にダウングレードされています。

認証を受けるには LDAP バインド操作を使用します。バインドは、ユーザの認証情報に基づいて ID を作成します。TLS を停止するとき、LDAP サービスは以前に確立された認証をすべて削除します。認証ステータスが匿名に変わります。匿名以外の状態に切り替える場合は、再認証を受ける必要があります。

**シナリオ : 再認証を受ける** — あるユーザが STOPTLS を実行します。すると、そのユーザの状態が匿名に変わります。このユーザのファイルにネット上でアクセスするには、Bind コマンドを実行し、ログイン認証情報を入力します。ユーザが認証され、インターネット上でクリアテキストで作業を続行できます。

## TLS のサーバを環境設定する

TLS セッションがインスタンス化されると、ハンドシェークが行われます。サーバとクライアントがデータを交換します。ハンドシェークの方法はサーバが決定します。サーバの正当性を証明するため、サーバは常にサーバの証明書をクライアントに送信します。このハンドシェークにより、そのサーバがクライアントに指定されたサーバであることが証明されます。

クライアントにも正当性の証明を要求するには、サーバに値を設定します。これは `LdapTLSVerifyClientCertificate` という属性です。

値	説明
0	オフハンドシェーク時に、サーバがクライアントに証明書を提供します。サーバがクライアントに証明書の送信を要求することはありません。クライアントは証明書を使用することも、無視することもできます。セキュリティ保護されたセッションが確立されます。
1	ハンドシェークの間、サーバは証明書をクライアントに送信し、クライアントからも証明書を要求します。クライアントが証明書の返信を選択すると、その証明書が確認されます。サーバがクライアントの証明書を確認できない場合、接続は切断されます。  クライアントが証明書を送信しない場合、サーバは接続を維持します。
2	ハンドシェークの間、サーバはクライアントから証明書を要求します。クライアントが証明書を提供しない、または証明書が確認できない場合には、接続は切断されます。

サーバが TLS をサポートするよりも前に、サーバにその正当性の証明に使用する X.509 証明書を提供する必要があります。

この証明書は、eDirectory のインストール時に自動的に提供されます。インストール時に、パブリックキー暗号化サービス (PKI) で、キーオブジェクトと Novell Modular Authentication Services(NMAS™) が作成されます。次の図は、iManager におけるこれらのオブジェクトを示しています。



インストール中、これらの証明書の 1 つが LDAP サーバと自動的に関連付けられます。Novell iManager の LDAP サーバオブジェクトの [接続] タブに DN が表示されます。この DN は、X.509 の証明書を表しています。次の図のサーバ証明書フィールドは、この DN を示しています。



Novell iManager で、暗号化キーオブジェクト (KMO) 証明書を参照できます。また、ドロップダウンリストから、別の証明書に変更することもできます。DNS または IP 証明書のいずれかを使用します。

検証の際には、サーバは証明書にある名前(ハード IP アドレスまたは DN)を確認します。

TLS 接続を確立するには、次の条件を確認します。

- ◆ LDAP サーバはそのサーバの KMO を認知する必要があります。
- ◆ クリアポートに接続してから、セキュアポートに接続するか TLS を開始します。

LDAP サーバを再設定し、サーバをリフレッシュします。309 ページの「LDAP サーバをリフレッシュする」を参照してください。ConsoleOne と Novell iManager は、自動的にサーバをリフレッシュします。

## TLS のクライアントを環境設定する

LDAP クライアントとは、たとえば Netscape Communicator、Internet Explorer、ICE のようなアプリケーションです。クライアントは、LDAP サーバが使用する認証局を認知している必要があります。

サーバが eDirectory ツリーに追加されると、インストールの際にデフォルトで次が作成されます。

- ◆ ツリーの認証局 (ツリー CA)
- ◆ ツリー CA からの KMO

LDAP サーバはこの認証プロバイダを使用します。

クライアントは、LDAP サーバが使用していると主張するツリー CA を確認できるように、信頼する証明書をインポートする必要があります。この証明書をサーバからインポートしておく、サーバがその証明書を送信してきたときに、クライアントはそれを確認し正当なサーバであるかどうか確かめることができます。

クライアントが安全に接続できるように、接続前にクライアントの環境設定をしておく必要があります。

クライアントによる証明書のインポート方法は、使用しているアプリケーションの種類によって異なります。各アプリケーションには、何らかの証明書をインポートする方法があります。Netscape ブラウザ、IE、ICE は、それぞれ別々の方法でインポートを行います。これらのインポート手段は、3 つの異なる LDAP クライアントです。各クライアントは、それぞれの方法で信頼する証明書を探します。

## ルート認証局をエクスポートする

ルート認証局は、証明書サーバを受け入れるときに自動的にエクスポートできます。

ルート認証局を手動でエクスポートするには、『[Exporting a Trusted Root or Public Key Certificate \(http://www.novell.com/documentation/japanese/crt221ad/index.html\)](http://www.novell.com/documentation/japanese/crt221ad/index.html)』を参照してください。

エクスポート機能により、指定したファイルが作成されます。ファイル名は変更できませんが、オブジェクトのタイプを認識できるように、ファイル名に「DNS」または「IP」を残しておくことをお勧めします。また、サーバ名も残しておきます。

eDirectory との安全な LDAP 接続を確立するブラウザに、自己割り当て認証局をインストールします。

Internet Explorer など、Microsoft 社の製品で証明書を使用する場合は、.der 拡張子を残しておくようにします。


アプリケーションまたは SDK が証明書を要求する場合は、証明書をデータベースにインポートします。

Internet Explorer 5 の場合は、ルート証明書が自動的にエクスポートされ、レジストリが更新されます。これには、Microsoft が通常使用している .x509 拡張子が必要です。

## クライアント証明書で認証を受ける

相互認証には、TLS セッションとクライアント証明書が必要です。サーバとクライアントの両方が、それぞれが自分の主張するオブジェクトであることを証明する必要があります。クライアントの証明書がトランスポート層で確認されます。しかし、LDAP プロトコル層では、LDAP バインド要求を出すまでクライアントは匿名になります。

この時点で、クライアントはその正当性をサーバに証明しましたが、LDAP にはまだ証明されていません。クライアント証明書に含まれた ID で認証を受けたい場合は、クライアントは SASL EXTERNAL メカニズムを使ってバインドされます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] の順にクリックします。
- 3 [LDAP サーバの表示] をクリックし、LDAP サーバオブジェクトの名前をクリックします。
- 4 [接続] をクリックします。
- 5 トランスポート層のセキュリティセクションのドロップダウンメニューから [クライアント証明書]、[必須] の順に選択します。  
これにより、相互認証が可能になります。
- 6 [適用] をクリックし、[OK] をクリックします。

## サードパーティプロバイダの証明書を使用する

eDirectory のインストール時に、LDAP サーバにツリー認証局 (CA) が提供されます。LDAP キーオブジェクトは、その CA に基づいています。クライアントが LDAP サーバに送信する証明書は、このツリー CA から確認できます。

LDAP Services for eDirectory 8.7.3 は、複数の認証局をサポートしています。Novell のツリー CA はそのうちの 1 つです。LDAP サーバが、他の CA を持っている場合があります (例: 外部団体の VeriSign\* など) この追加 CA もルート認証局です。

LDAP サーバが複数の認証局を使用するように環境設定するには、LDAP サーバオブジェクトの ldapTLSTrustedReaderContainer 属性を設定します。LDAP サーバが複数の認証局を参照することにより、クライアントは外部の認証局を使用できます。

## LDAP プロキシユーザを作成および使用する

Novell eDirectory は、認証されていないユーザに [Public] 識別子を割り当てます。LDAP プロトコルでは、認証されていないユーザは匿名ユーザになります。デフォルトでは、LDAP サーバは匿名ユーザに [Public] 識別子の権利を与えます。この権利により、非承

認の eDirectory および匿名 LDAP ユーザは、[Public] 権を使用して eDirectory を参照することができます。

また、LDAP サーバは匿名ユーザによる別のプロキシユーザの権利の使用を許可します。この値は LDAP グループオブジェクトにあります。Novell iManager では、この値は、[プロキシユーザ] フィールドで指定します。また ConsoleOne の場合は、[プロキシユーザ名] フィールドで指定します。次の図は、Novell iManager のフィールドを示しています。




プロキシユーザは識別名です。このプロキシ ID に、Public 識別子とは別の権利を与えることができます。プロキシユーザを使用すると、eDirectory ツリー内の特定コンテナへの LDAP 匿名アクセスを制御できます。

**注:** プロキシユーザのログイン制限は、すべての匿名 LDAP ユーザに適用する場合以外は設定しないでください。

**シナリオ : NLDAP プロキシユーザを設定する** — Digital Airlines 社はリサーチ会社の DataSure 社と契約を締結しています。DataSure 社は LDAP を使用して、Digital Airlines 社の NetWare 6 サーバである DigitalAir43 にアクセスし、そのリサーチを保存します。Digital Airlines 社は、DataSure 社に DigitalAir43 のディレクトリに対する [Public] 権を与えたくないとしています。

そのために、LDAP プロキシユーザを作成し、そのユーザに DataSure ディレクトリに対する特定の権利を割り当てます。LDAP グループオブジェクトにプロキシ識別名を作成し、サーバをリフレッシュします。サーバは自動的に、すべての新規または既存の匿名ユーザについて、プロキシユーザの権利の使用を開始します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの作成] の順にクリックし、プロキシユーザを作成します (例 :LDAPProxy)。
- 3 ユーザに NULL パスワードを割り当てます。
- 4 (オプション) 指定したディレクトリにプロキシユーザの権利を割り当てます。
- 5 [LDAP] > [LDAP の概要] > [View LDAP グループの表示] > [LDAP グループオブジェクト] の順にクリックします。
- 6 [プロキシユーザ] フィールドで [参照] ボタンをクリックし、LDAPProxy ユーザを選択して [OK] をクリックします。

## SASL を使用する

SASL (Simple Authentication and Security Layer) は、IANA (Internet Assigned Numbers Authority) で登録する必要のあるさまざまな認証メカニズムを定義します。LDAP サーバでは次のメカニズムがサポートされます。

- ◆ DIGEST-MD5
- ◆ EXTERNAL
- ◆ NMAP\_LOGIN

これらのメカニズムは、eDirectory のインストールまたはアップグレード時に、サーバにインストールされます。ただし、UNIX の場合は `nmapinst` ユーティリティを実行して NMAP メソッドをインストールする必要があります。

LDAP サーバは、SASL に問い合わせる環境設定時にインストールしたメカニズムを検索し、インストールされたメカニズムを自動でサポートします。また、`supportedSASLMechanisms` 属性を使って `rootDSE` で現在サポートされている SASL メカニズムをレポートします。

これらのメカニズムは登録されているため、名前はすべて大文字で入力します。大文字で入力しない場合、LDAP サーバはメカニズムを認識しません。

LDAP バインドプロトコルでは、クライアントは認証に様々な SASL メカニズムを使用することができます。アプリケーションが LDAP バインド API を使用している場合は、単純バインドを選択し、DN とパスワードを入力するか、SASL バインドを選択し、SASL メカニズム名 (大文字) と、そのメカニズムが要求する、関連する SASL 認証情報を提供する必要があります。

### DIGEST-MD5

DIGEST-MD5 メカニズムには TLS は必要ありません。LDAP サーバは、クリア接続およびセキュア接続の両方で DIGEST-MD5 をサポートします。

LDAP は、バインド要求で SASL メカニズムをサポートします。ユーザが LDAP 単純バインド (DN およびクリアテキストパスワード) の代わりに、LDAP SASL バインドを要求すると、DN と MD5 認証情報が提供されます。

MD5 は、暗号化されたパスワードのハッシュを提供します。パスワードは、クリア接続でも暗号化されます。そのため LDAP サーバは、ポートがクリアテキストポートか暗号化されたポートかに関係なく、MD5 を使ったパスワードを受け入れます。

このパスワードが第三者によって見破られることはありません。ただし、接続全体に対してなりすましやハイジャックがなされる可能性はあります。

メカニズムは LDAP SASL バインドです (単純バインドではありません)。そのため、インストール時に [パスワードとの単純バインドに TLS を必要とする] チェックボックスがオンになっていても、LDAP サーバはこの要求を受け入れます。

### EXTERNAL

EXTERNAL メカニズムにより、ユーザの DN および認証情報がサーバに提供されたことが LDAP サーバに通知されます。そのため、バインド要求時には DN と認証情報は必要ありません。

SASL EXTERNAL メカニズムを使用した LDAP バインド要求は、サーバに次を実行するよう要求します。

- ◆ EXTERNAL 層に認証情報を問い合わせる
- ◆ ユーザをその認証情報を持つユーザとして認証する



安全なハンドシェイクが行われます。サーバはクライアントから認証情報を要求し、クライアントはそれをサーバに渡します。LDAP サーバはクライアントから渡された証明書を受け取り、それを NMAS モジュールに渡して、ユーザを証明書にある DN として認証します。

使用できる DN の証明書を使用するには、クライアントを設定する必要があります。証明書の設定に関する詳細は、[NMAS \(http://www.novell.com/documentation/japanese/nmas22/index.html\)](http://www.novell.com/documentation/japanese/nmas22/index.html) オンラインマニュアルを参照してください。

クライアントが EXTERNAL メカニズムを送信しても、LDAP サーバは要求の処理に失敗することがあります。Novell iMonitor により、処理が失敗した原因が通知されます。次のような原因が考えられます。

- ◆ 接続がセキュア接続ではない。
- ◆ 接続はセキュア接続だが、クライアントがハンドシェイク時に要求された証明書を提供しなかった。
- ◆ SASL モジュールを使用できない。
- ◆ クライアントが要求を送信する前に rootDSE をチェックしなかった。

## NMAS\_LOGIN

NMAS\_LOGIN メカニズムにより、LDAP サーバに NMAS のバイOMETリック機能が提供されます。詳細については、Novell NDK を参照してください。

サーバの起動時に、LDAP サーバは SASL モジュールを初期化し、その LDAP サーバがどのメカニズムを使用できるかを SASL モジュールに問い合わせます。

クライアントは rootDSE に問い合わせ、サポートされているメカニズム属性を検索することができます。LDAP サーバはその後、サポートされているメカニズムを表示します。

## LDAP サーバを使ってディレクトリを検索する

このセクションでは、次の情報について説明します。

- ◆ [317 ページの「検索制限を設定する」](#)
- ◆ [318 ページの「参照を使用する」](#)
- ◆ [324 ページの「フィルタ済みレプリカを検索する」](#)

### 検索制限を設定する

次の LDAP サーバオブジェクトの属性により、LDAP サーバのディレクトリ検索方法を指定することができます。

- ◆ 検索エントリの制限

検索のサイズを制限します。デフォルトは 0 で、サイズの制限はありません。LDAP サーバの負荷が大きくなりすぎないように、検索要求に対して LDAP サーバが返すエントリ数を制限できます。


**シナリオ：検索のサイズを制限する** — ユーザは、検索結果が何千件にもなりそうな、あるオブジェクトの検索を要求します。ただし、検索結果は 10 件に制限してあります。LDAP サーバは 10 件の検索結果を返すと検索を中止します。一致するデータがまだ存在しているが、検索が終了されたことを告げるシステムメッセージが表示されます。

- ◆ 検索時間制限

サーバが検索を行う時間を制限します。デフォルトは 0 秒です。これは時間制限がないことを表します。

次の図は、Novell iManager におけるこれらの属性を示します。



- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] > [LDAP サーバの表示] の順にクリックします。
- 3 [LDAP サーバオブジェクト] > [検索] の順にクリックします。
- 4 制限セクションをスクロールして値を入力し、[OK] をクリックします。

クライアントも、検索要求に制限を設定することもできます (たとえば検索を 2 秒に制限するなど)。クライアントの制限がサーバの制限と競合する場合、LDAP サーバは値が小さい方の要求を採用します。

検索は、アクセス制御リスト (ACL) に基づいて実行されます。このため、匿名検索の場合は、ディレクトリに何千というエントリが存在していても、Public 権で見ることのできるごく一部のエントリしか返されることがあります。

## 参照を使用する

参照は、名前を解決するためのクライアント中心の方法です。LDAP クライアントが LDAP サーバにリクエストを送信すると、LDAP サーバはリクエスト操作のターゲットエントリをローカルに見つけようとします。LDAP サーバはターゲットエントリを見つけられないと、所有する知識参照を使用して、そのエントリについてのより多くの知識を持つ第 2 の LDAP サーバへの参照を生成します。第 1 のサーバは、参照情報を LDAP クライアントに送信します。

次に、LDAP クライアントは第 2 の LDAP サーバへ接続し、操作を再実行します。第 2 の LDAP サーバが操作のターゲットエントリを保持している場合は、そのサーバが操作を実行します。エントリを保持していない場合は、第 2 の LDAP サーバもまた参照をクライアントに送信します。この操作は、次のいずれかの状況になるまで続けられます。

- ◆ クライアントがターゲットエントリを保持するサーバと接続し、要求する操作が実行できる。
- ◆ LDAP サーバが、エントリは存在しないとのエラーを返す。
- ◆ LDAP サーバがこれ以上参照がないことを通知する。

LDAP for eDirectory 8.7 で導入された機能により、参照の動作が以前のバージョンの eDirectory および NDS から少し変更されました。これにより、LDAP サービスの環境設定方法も変更されました。

## デフォルトの参照

通常、デフォルトの参照 URL には、ツリーのルートを保持するサーバを指す LDAP URL が含まれています。LDAP URL の形式を次に示します。ldap://host:port

[デフォルト参照 URL] フィールドに、デフォルトの参照を入力します。



これまで、eDirectory LDAP サーバは、多くのフェールオーバー時にデフォルトの参照を送信していました。これが、この動作を予期しないユーザを混乱させる結果になっています。そこで、LDAP Services for eDirectory 8.7.3 では、サブオーディネート参照で、デフォルトの参照が送信される場合を指定できるようになりました。

この新しいオプションは、LDAP サーバおよび LDAP グループオブジェクトの ldapDefaultReferralBehavior 属性の値で指定します。値は次のビットのビットマスクである整数です。

ビット	値
0x00000001	ベース DN が見つかりません
0x00000002	ベース DN は、利用できない eDirectory サーバ上にあります
0x00000004	検索スコープのエントリは、利用できない eDirectory サーバ上にあります

LDAP サーバがその操作に対して「常に参照する」に設定されており、リストされたいずれかの条件と一致し、対応する値が設定されている場合、デフォルトの参照が返されます。

## 検索操作の参照を設定する

LDAP for eDirectory 8.7 で導入された機能により、参照の動作が以前のバージョンの eDirectory および NDS から少し変更されました。これにより、LDAP サービスの環境設定方法も変更されました。

eDirectory ツリー内の他の eDirectory サーバに参照を返すように eDirectory LDAP サーバを設定することができます。デフォルトでは、LDAP サーバはユーザに代わってすべての操作を他の eDirectory サーバにチェーンし、参照は返しません。

eDirectory 8.7 より以前は、参照オプションの設定は LDAP グループオブジェクトでだけしか使用できませんでした。eDirectory 8.7.3 では、LDAP サーバオブジェクトにもこのオプションを設定することができるようになりました。LDAP サーバオブジェクトの設定により、LDAP グループオブジェクトの設定は上書きされます。

dapSearchReferralOption 属性を操作することにより、参照オプションを設定することができます。LDAP Services for eDirectory 8.7 より以前は、この属性を次のオプションに設定することができました。

- ◆ 321 ページの「チェーンを優先する」(デフォルトオプション)
- ◆ 322 ページの「参照を優先する」
- ◆ 322 ページの「常に参照する」

これらの参照オプションは、eDirectory ツリー内の他の eDirectory サーバの参照およびチェーンでだけ使用できます。この設定は、非委任パーティションからの参照は制御しません。そのため、[参照オプション] ドロップダウンリストでオプション ([常にチェーンする] など) を選択しても、他のサーバの非委任パーティションからは参照が送信されます。

LDAP Services for eDirectory 8.7.a では [常にチェーンする] オプションにより、eDirectory DSA 以外の上方向参照がサポートされています。321 ページの「常にチェーンする」を参照してください。

次の図は、検索およびその他の操作に使用する [LDAP 参照] ドロップダウンリストを示しています。



eDirectory 操作にはこの他に、「追加」、「削除」、「編集」、「バインド」の各操作の参照があります。

## 常にチェーンする

[常にチェーンする] オプションは、「まったく参照しない」ように設定するオプションです。このオプションを選択すると、eDirectory LDAP サーバは、eDirectory ツリー内にある他の eDirectory サーバに参照を返しませんが、LDAP サーバは、要求を出したクライアントに代わって他の LDAP サーバをチェックし、クライアントに参照を返します。

[常にチェーンする] オプションは、eDirectory をグローバル連結ツリーの従属サーバとして使用している場合に適しています。

この参照オプションは、eDirectory ツリー内の参照の処理設定にのみ使用します。このオプションが eDirectory サーバ以外のサーバの参照の動作に影響することはありません。

他の eDirectory サーバへの参照をブロックすることによりあまり意味はありませんが、これが重要になる場合もあります。eDirectory 8.7 以降のサーバ上の非委任データを古いバージョンの eDirectory サーバ上で複製すると、古いサーバを参照したときにクライアントアプリケーションのグローバルツリーが歪んで表示されることがあります。

たとえば、LDAP クライアントは LDAP サーバの参照をキャッシュし、最後に通信したサーバに要求を送信するものとします。クライアントが上方参照をサポートする eDirectory サーバに要求を送信するよう設定されている場合、クライアントのグローバルツリーは正常に表示されます。

しかし、eDirectory 8.7 以前の LDAP サーバは、非委任領域と上方参照を認識しません。このため、クライアントが eDirectory ツリー内の以前のバージョンの eDirectory サーバの参照に従い、要求をそのサーバに送信し続けると、以前のバージョンの LDAP サーバにより、非委任データが実際のディレクトリツリーデータであるかのように表示されてしまいます。

ただし、クライアントによっては、RootDSE の supportedFeatures 属性をチェックし、サーバが上方参照をサポートしているか確認できるものもあります。

## チェーンを優先する

[チェーンを優先する] オプションを選択すると、通常、検索操作で参照は返されませんが、LDAP サーバはその代わりに、すべての eDirectory DSA に対する検索操作を実行します。

ただし、持続的検索制御を設定して検索を実行する場合は例外となります。Novell の実装する持続的検索ではチェーンがサポートされていないため、検索スコープがローカルに限定されていない場合に参照が送信されます。

LDAP サーバが検索操作を受信します。ツリーのエントリがローカルに格納されていない場合、サーバは自動的に他のサーバにチェーンします。エントリの検出後、LDAP サーバは LDAP クライアントのプロキシとして機能します。LDAP サーバは LDAP クライアントにバインドされたものと同じ識別情報を使用してリモートサーバの認証を受け、そこで検索操作を続行します。

最初に要求を受信した LDAP サーバが、LDAP クライアントにすべての検索エントリと検索結果を送信します。この LDAP サーバが要求をすべて処理するため、LDAP クライアントからは他のサーバが関与していることはわかりません。

eDirectory でチェーンを使用すると、ある LDAP サーバに多くのデータがない場合でも、そのサーバがツリー全体のデータを保持しているかのように見えます。

[チェーンを優先する] は、パーティションに深くかかわるオプションです。

**シナリオ:他のパーティションで情報を探す** — Digital Airlines 社で、ユーザが LDAP サーバ DAir43 に [チェーンを優先する] オプションを選択しました。DAir43 はパーティション A にあります。パーティション B は A のサブパーティションで、LDAP サーバ DAir44 はこのパーティションにあります。

ある LDAP クライアントが検索を要求します。DAir43 は、エントリーをローカルで検索しますが、データは一部しか見つかりません。DAir43 は、要求されたエントリーを持つ DigitalAir44 に自動的にチェーンします。DAir44 は、DAir43 にデータを送信し、DAir43 は、LDAP クライアントにエントリーを送信します。

[チェーンを優先する] オプションを使用すると、操作が持続的検索である場合を除き、LDAP サーバは必要に応じて検索要求を他のサーバにチェーンします。持続的検索の詳細については、[329 ページの「持続的検索 :eDirectory イベントの設定」](#)を参照してください。

### 参照を優先する

[参照を優先する] オプションを選択すると、必要に応じ、参照の検索結果が eDirectory ツリー内の他の eDirectory サーバに返されます。この参照は、データを持つサーバが動作可能であり、LDAP サービスが稼働していることをローカルサーバが確認した場合のみ送信されます。それ以外の場合、操作は他のサーバにチェーンされるか、他のサーバが動作していない場合は処理に失敗します。

パーティションが 2 つあり、サブツリー検索を実行するとします。ローカルサーバから検索エントリーがすべて検出されるまで検索が実行されます。次に、他のサーバの検索が実行されます。データのレプリカ (そのパーティション) を持つサーバが nldap.nlm も実行している場合、LDAP サーバは LDAP 参照を作成し、それを LDAP クライアントに返します。

レプリカのあるサーバが nldap.nlm を実行していない場合、LDAP サーバは要求を他のサーバにチェーンし、そこで検索を完了します。

nldap.nlm が起動されると、LDAP サーバはその LDAP サーバが参照先になっている eDirectory と通信します。クライアントが参照を受信したが、その参照が停止した場合は、LDAP サーバが実行されていません。

### 常に参照する

[常に参照する] オプションは、デフォルト参照がさまざまなフェールオーバー (たとえば、オブジェクトが見つからなかったり、サーバがダウンしているなど) の状況で送信される場合を除き、[参照を優先する] と同じロジックに従います。

残りのデータを保持している他のサーバで LDAP サービスが実行されていない場合、最初の LDAP サーバは要求を第 2 のサーバへチェーンしません。

[常に参照する] オプションを設定している場合には、デフォルト参照を指定することができます。[デフォルトの照会先] フィールドで 2 つの異なるベンダの LDAP サーバを結合し、独自のディレクトリツリーを構築することができます。

**シナリオ:デフォルトサーバを使用する** — 1 つの LDAP ツリーがあるとします。ツリーの一部には eDirectory のサービスが適用されています。従属パーティションには iPlanet のサービスが実行されています。[デフォルトの照会先] フィールドに、iPlanet サーバの URL を入力します。ある LDAP クライアントが検索を要求します。

ベース DN を解決できないため、LDAP サーバは [デフォルトの照会先] フィールドに入力された文字列をクライアントに送ります。LDAP クライアントはこの参照の URL で指定された場所を参照して iPlanet サーバに接続し、ここで検索は完了します。

デフォルト参照が設定されており、ベース DN が見つからない場合、サーバはクライアントにデフォルト参照を返します。

参照の形式は、LDAP URL です (例 : LDAP://123.23.45.6:389)。

LDAP サーバは、デフォルト参照をクライアントに送信する際 (ベース DN が利用できない場合)、スラッシュ (/) とクライアントが検索中の DN を追加します。デフォルト参照と追加された情報がクライアントに送信されます。クライアントはデフォルト参照で指定したサーバに検索要求を送信します。

LDAP グループオブジェクトには、デフォルト参照の文字列フィールドがあります。LDAP サーバは、そのデータを文字列として扱います。このとき、確認は行われません。入力された文字列が、参照の先頭に追加されます。また、なんらかのデータが参照に追加されます。LDAP サーバが受け入れる文字列は、URL のような形式になります。

LDAP が実行されている他の eDirectory サーバの参照がクライアントに返される時、クライアントは 1 つのサーバにつき 2 つの参照を受信します。

- ◆ クライアントをクリアテキストポートに導く参照
- ◆ クライアントをセキュアポートに導く参照

2 つの参照を区別するために、クリアテキスト参照には ldap://、セキュアポート参照には ldaps:// が付きます。

サーバからの参照の場合は、ポート番号を追加します。

## 他の操作の参照を設定する

履歴参照オプション設定は、検索操作にのみ使用します。他の処理に比較オプションを適用する場合は、ldapOtherReferralOption 属性が使用されます。この属性により、同じ値を使って検索以外の操作の動作を制御できます(参照を送信しないバインドは除きます)。

## ManageDsaIT の非サポート

LDAP Services for eDirectory 8.7.3 では、eDirectory ツリー内の eDirectory サーバの分散関係は、ManageDsaIT 制御以外の方法で管理されます。LDAP クライアントは、ManageDsaIT 制御を使って eDirectory 従属または相互参照の問い合わせや更新を実行することはできません。

## サポートされていない機能

LDAP Services for eDirectory 8.7.3 では、サブオーディネートリファレンスはサポートされていません。委任パーティションの従属パーティションとして非委任パーティションを作成したり、そのパーティションから参照を送信させると失敗する場合があります。これを行う場合、参照は操作のベース DN を解決するときのみ送信されます。SearchResultReferences は送信されません。

非委任領域のデータの分散更新はサポートされていません。ルートサーバで名前の変更があった場合、非委任領域で同じデータを持った eDirectory サーバに名前の変更をコピーするようなメカニズムは組み込まれていません。

## フィルタ済みレプリカを検索する

フィルタはレプリカが持つデータ量を制限します。そのため、フィルタ済みのレプリカには、ディレクトリが保持する実データが完全には表示されません。次はレプリカに適用されたフィルタの例です。


- ◆ レプリカに含まれるのはユーザオブジェクトだけです。
- ◆ レプリカにはすべてのユーザオブジェクトが含まれますが、オブジェクトには電話番号と住所しか含まれません。

フィルタ済みレプリカのデータは不完全なため、LDAP 検索の結果も制限されます。そのため、デフォルトでは、LDAP 検索要求はフィルタ済みレプリカを調べません。

次のような場合は、フィルタ済みレプリカ検索を実行しても、レプリカフィルタから何も検索結果が返されないことがあります。

- ◆ 検索フィルタに一致するオブジェクトがローカルのフィルタ済みレプリカサーバに存在しない場合、結果が完全なレプリカサーバから取得されるため、ローカルのレプリカフィルタの結果と一致しないことがあります。
- ◆ 検索ベースがフィルタ済みレプリカサーバのローカルにない場合、検索フィルタに一致するオブジェクトが完全なレプリカサーバから取得され、これがローカルレプリカのフィルタの結果と一致しないことがあります。

ただし、フィルタ済みレプリカに必要なデータがあることがわかっている場合は、LDAP サーバがフィルタ済みレプリカを検索するように設定することができます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] の順にクリックします。
- 3 [LDAP サーバの表示] をクリックし、LDAP サーバの名前をクリックします。
- 4 [検索] をクリックします。
- 5 [検索にフィルタ済みレプリカに含める] を選択し、[適用] をクリックします。



## 上方参照を設定する

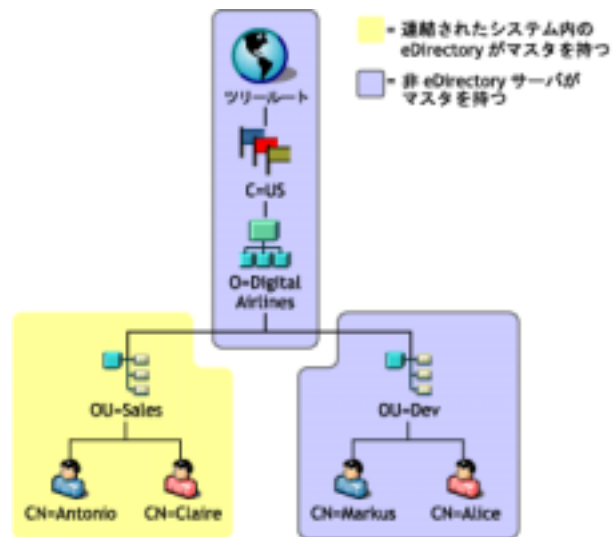
組織の規模が大きくなると、さまざまなベンダの LDAP サーバソフトウェアを使用したディレクトリツリーが必要になります。このようなツリーを、グローバル連結ツリーといいます。LDAP Services for eDirectory 8.7.3 には、参照を連結ツリーの上方の DSA に返す機能があります。



## シナリオ：連結ツリーでの上方参照

Digital Airlines 社には、あるネットワーク担当者がいます。Digital Airlines 社のディレトリツリーのルート(ツリーのルートから O=Digital Airlines まで)のマスタは、OpenLDAP サーバ上にあります。組織 (OU=Sales) のマスタは eDirectory サーバにあり、その他の組織 (OU=Dev) は iPlanet サーバ上にあります。

次の図は、このツリーを示します。



eDirectory にマスタがあるのは、OU=Sales のパーティション内のデータだけです。他の領域のデータのマスタは eDirectory 以外の DSA 上にあります。ネットワーク担当者は、操作の対象が O=Digital Airlines より上の領域、または OU=Sales 階層に属さない O=Digital Airlines より下の領域である場合、上方参照を返すように LDAP サービスを設定します。

ベース DN が OU=Dev, O=Digital Airlines, C=US の eDirectory LDAP サーバに操作が送信されます。そのエントリを保持するサーバ、またはそのエントリを保持するサーバを認知しているサーバを指す参照が返されます。

同様に、O=Digital Airlines, C=US が対象のサブツリー検索でも、ルート DSA の参照が返されます。ルート DSA は、OU=Sales および OU=Dev のマスタである DSA の参照を返します。

LDAP サービスにより、eDirectory サーバがこのツリーに参加するのにデータ階層データを「非委任」パーティションに持つことができます。非委任領域のオブジェクトに含まれるのは、正しい DN 階層を構築するのに必要なエントリだけです。これらのエントリは、X.500 の“Glue”エントリに類似しています。

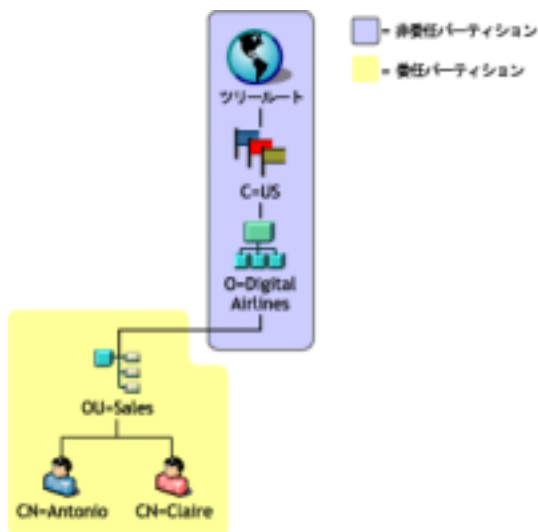
このシナリオでは、ルート、C=US、O=Digital Airlines オブジェクトは非委任領域の eDirectory サーバにあります。

eDirectory では、知識情報 (参照データ) を非委任領域に置くことができます。この情報は、LDAP クライアントに参照を戻すのに使用されます。

LDAP 操作が eDirectory ツリーの非委任領域で実行されると、LDAP サーバは正しい参照データを検索し、クライアントに参照を戻します。

## 非委任領域を作成する

次の図は、325 ページの「シナリオ：連結ツリーでの上方参照」で示した、連結ツリーの eDirectory サーバが保持する実際のデータを示しています。



エント리는、マスタが他の DSA 上にあっても、OU=Sales 上に配置されます。これは、eDirectory サーバが持つエントりに適切な DN を提供できるようにするためです。

非委任領域を作成するには、次を実行します。

- 1 非委任データを委任データから切り離します。

委任領域の最上部に、パーティション境界を作成します。他に指定がない限り、eDirectory サーバは自身が保持しているすべてのデータに対して委任されているとみなされます。

- 2 ルートパーティションを非委任としてマークします。

**2a** 委任属性を、パーティションの最もルートに近い属性に追加します。

**2b** 値が 0 の委任属性を作成します。

- 3 非委任領域の最下部に境界線をひきます。

このサーバが委任されるサブツリー領域にパーティションルートを作成します。たとえば上の図の場合は、パーティションルートは OU=Sales エントリにあります。新しいパーティションには、0 に設定された委任属性はありません。そのため、サーバはパーティションに対する委任を受けることになります。

- 4 LDAP サーバをリフレッシュします。

LDAP サーバは、その設定がリフレッシュされるたび、委任および非委任領域の境界をキャッシュします。手動でサーバの設定をリフレッシュしない場合、サーバは 30 分毎のバックグラウンドタスクで自動的にリフレッシュされます。

複数のパーティションがある場合は、非委任領域のチェーンに重ねることができません。ただし、LDAP Services for eDirectory 8.7.3 では、すべての非委任パーティションは連続していなければならない、ローカルレプリカがそれを保持している必要があります。

## 参照データを指定する

操作が非委任領域で実行されていることが検出されると、LDAP サーバは参照をクライアントに返すために使用できる情報を探します。この参照情報は、次のいずれかになります。

- ◆ 非委任領域のいずれか、またはすべてのエントリにある情報
- ◆ サーバの設定データを持つLDAPサーバまたはLDAPグループオブジェクト上の、デフォルト参照として指定された情報

非委任領域のエントリが持つ参照情報は、即時上方参照です。このような参照情報は、複数の値をとる `ref` 属性から構成されています (この属性の詳細については、[RFC 3296 \(http://www.ietf.org/rfc/rfc3296.txt\)](http://www.ietf.org/rfc/rfc3296.txt) を参照してください)。デフォルト参照設定が持つ参照情報は上方参照で、値を1つとります。(X.501 の `immSupr` および `supr DSE` タイプを参照してください)。

参照データは LDAP URL の形式で保持されますが、これにはホストと (オプションで) 参照先の DSA のポートだけしか指定されていません。この参照データの例は次のようになります。

```
ldap://ldap.digital_airlines.com:389
```

LDAP サーバは操作のベース DN (見つからない場合は一致した DN) を参照します。ベース DN に参照情報が含まれる場合、LDAP サーバはその情報を参照として返します。

参照情報が見つからない場合、LDAP サーバはツリーの上方向に向かって参照情報を探します。すべてのエントリを検索しても参照情報が見つからない場合、LDAP サーバは上方参照を返します (この参照は、LDAP グループまたは LDAP サーバオブジェクト上のデフォルト参照設定で保持されます)。

### 即時上方参照の追加

`immediateSuperiorReference` と呼ばれる補助オブジェクトクラスを非委任領域のエントリに追加することができます。この補助クラスは、1つ以上の LDAP URL とともに作成される `ref` 属性を追加します。それぞれの URL は DSA のホスト名と (オプションで) ポートを指します。

### 上方参照を追加する

これまで、LDAP グループオブジェクトは `ldapReferral` 属性を持っていました。この属性は、eDirectory ツリーの他の eDirectory サーバに参照を返すときに発生する、さまざまなフェールオーバーの状況で使用されるデフォルトの参照を保持していました。LDAP Services for eDirectory 8.7.3 では、この属性を使用して連結ツリーの上方向 DSA のデフォルト参照を1つ指定します。

また、`ldapReferral` 属性が LDAP サーバオブジェクトに追加されました。`ldapReferral` 属性に LDAP サーバオブジェクトの値が含まれる場合、この設定により LDAP グループオブジェクトの同じ属性の値は上書きされます。この動作により、グループに属するすべての LDAP サーバに特定のデフォルト参照を設定し、1つか2つのサーバのデフォルト参照を別のデフォルトで上書きすることができます。

`ldapReferral` 属性の値は、LDAP URL です。URL には、ホストと参照先の DSA のポート (オプション) が含まれます。

## LDAP で参照情報を更新する

上記のステップに従い、LDAP を使ってこのタスクを実行しても、即時上方参照を追加することはできないことが多くありました。これは、ルートパーティションが既に非委任とマークされており、LDAP はパーティション内のデータに対するどのような操作の参照も送信していたためです。

非委任領域の情報の更新または問い合わせを行うには、LDAP 要求に ManageDsaIT 制御を設定する必要があります。この制御の詳細については、RFC 3296 (<http://www.ietf.org/rfc/rfc3296.txt>) を参照してください。この制御により、LDAP サーバは非委任領域全体を委任領域であるかのように扱うことができます。

注: 上方参照機能は、LDAP でのみ利用できます。他のプロトコル (NDAP など) には、委任属性が存在することによる影響はありません。そのため、ConsoleOne または Novell iManager を使用した非委任領域のデータの参照や更新が妨害されることはありません。

## 影響を受ける操作

非委任領域と上方参照は、次の LDAP 操作に影響します。

- ◆ 検索と比較
- ◆ 編集と追加
  - DN 構文属性値はチェックされません。そのため、グループメンバー属性は、非委任領域のエントリを指す DN を含むことができます。
- ◆ 削除
- ◆ 名前の変更 (moddn)
- ◆ 移動 (moddn)
  - ペアレント DN が非委任領域にある場合、affectsMultipleDSAs エラーが返されます。
- ◆ 拡張

## 上方参照のサポートの有無を確認する

上方参照は、Novell LDAP Services for eDirectory 8.7 以降でのみサポートされています。ルート DSE の supportedFeatures 属性により、eDirectory サーバがこの機能をサポートしているかどうかを確認できます。supportedFeatures 属性の値が OID 2.16.840.1.113719.1.27.99.1 である場合は、この機能はサポートされています。その他のルート DSE オブジェクトに対する確認方法では、次が変更されています。

- ◆ namingContexts
  - この属性は、サーバが委任されているローカル DSA に保持されるパーティションルートだけを表示します。非委任パーティションルートは表示されません。
- ◆ altServer
  - この属性は、ローカルサーバと非委任パーティションだけを共有する他の eDirectory サーバを表示しません。
- ◆ superiorReference
  - この属性は、DSA の上方参照を通知します。この値は、LDAP サーバまたは LDAP グループオブジェクト上の ldapReferral 属性を更新することにより管理されます。

## 持続的検索 :eDirectory イベントの設定

Novell eDirectory では、ディレクトリ内で重大なイベントが発生したときにアプリケーションに通知するためのイベントサービスが用意されています。これには、ディレクトリサービスに関する一般イベントも含まれます。それ以外は eDirectory とその機能に固有のイベントです。

eDirectory イベントは、LDAP プロトコルに対して、次の 2 つの異なる拡張を通してアプリケーションに通知されます。

- ◆ 持続的検索制御の実装

Novell eDirectory の持続的検索機能は、最初の一致するエントリが返された後も続行される検索操作です。持続的検索は LDAP v3 の検索操作が拡張されたもので、クライアントからサーバへの検索結果内で更新をチェックする作業が不要になります。持続的検索制御により、クライアントは、ベース DN、検索スコープ、検索フィルタなどを指定する通常の LDAP 検索操作を実行することができます。その後、サーバは最後に SearchResultDone メッセージを返すのではなく、操作による接続が維持されます。このため、クライアントは検索結果に含まれるのエントリが変更されるたびに、最新のエントリを受け取ることができます。これにより、更新が発生するたびにクライアントは目的のエントリのキャッシュを維持したり、何らかのロジックをトリガすることができます。

この拡張についての詳細は、“Persistent Search” document on the Internet (<http://www.ietf.org/proceedings/01mar/I-D/ldapext-psearch-03.txt>) を参照してください。

- ◆ イベントの監視 (eDirectory 独自の拡張 LDAP 操作)



eDirectory イベントサービスを使用するアプリケーションは、ディレクトリに対する大きな計算負荷となることがあります。そこで、さまざまな属性パラメータを使用して、個々の eDirectory サーバにおけるイベントサービスの使用を制御することができます。これらのパラメータは LDAP サーバオブジェクト上に格納されます。パラメータを設定するには、ConsoleOne または Novell iManager を使用します。

イベントサービスを使用する特定のアプリケーションにより、これらのパラメータを特定の値に指定するよう要求される場合があります。特定のアプリケーション独自の要求は、そのアプリケーションのマニュアルに示されています。

詳細は、『Understanding and Using Persistent Search in Novell eDirectory(Novell eDirectory での持続的検索とその使用) (<http://developer.novell.com/research/appnotes/2003/february/04/a030204.htm>)』を参照してください。

## 持続的検索の管理

Novell iManager を使用すると、持続的検索を表示または編集することができます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 変更する LDAP サーバオブジェクトの名前とコンテキストを入力するか、 をクリックし、LDAP サーバオブジェクトを参照または検索します。



- 4 [OK] をクリックし、[全般] タブの [検索] をクリックします。



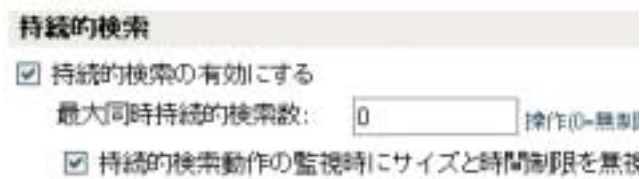
- 5 持続的検索を有効にします。

[持続的検索の有効化] チェックボックスはデフォルトでオンになっています。このサーバの持続的検索を無効にして禁止するには、チェックボックスをオフにします。

注：以前に確立された持続的検索操作を無効にすると、このオプションを無効にしてサーバをリフレッシュした後も操作が継続する場合があります。

- 6 このサーバ上の同時持続的検索の数を制御します。

[最大同時持続的検索数] フィールドの値を指定します。0 を指定すると、同時持続的検索数は無制限になります。




- 7 サイズおよび時間の制限を無視するかどうかを制御します。

持続的検索要求が最初の検索結果を送信した後で、サイズおよび時間制限を無視するかどうかを指定するには、[持続的検索動作の監視時にサイズと時間制限を無視する] チェックボックスをオンにします。

このオプションを選択しない場合、すべての持続的検索操作は検索制限の制約を受けます。サイズと時間のいずれかの制限に達した場合、検索操作は失敗し、該当するエラーメッセージが返されます。

- 8 [適用] をクリックし、[OK] をクリックします。

## イベントの監視拡張操作の使用を制御する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [LDAP] > [LDAP の概要] の順にクリックします。
- 3 [LDAP サーバの表示] をクリックし、LDAP サーバの名前をクリックします。

- 4 [イベント] をクリックします。



動作監視の有効化

最大動作監視負荷  (操作[0=無制限])

- 5 クライアントアプリケーションがこの LDAP サーバ上でイベントを監視できるかどうかを制御します。

クライアントアプリケーションがこの LDAP サーバ上でイベントを監視できるようにするには、[動作監視の有効化] チェックボックスをオンにします。

イベントの監視を無効にするには、チェックボックスをオフにします。

- 6 イベント監視アプリケーションがサーバ上に置くことのできる最大負荷を制御します。

[最大動作監視負荷] フィールドに値を入力します。

イベントデータの処理と、監視対象アプリケーションへのイベント通知の送信は、LDAP サーバに対して計算負荷となります。あるイベントによるサーバへの正確な負荷は、監視されるイベントの頻度、イベントに関係したデータ、およびそのイベントを監視しているクライアントアプリケーションの数によって決まります。

[最大動作監視負荷] は、イベント監視拡張がサーバにかけることができる負荷の大きさを示す相対値です。0 を指定すると、無制限になります。この属性の適切な値を見つけるには、実際にいろいろな値を試してみてください。

- 7 [適用] をクリックし、[OK] をクリックします。

## LDAP サーバの情報を取得する

LDAP サーバについての情報を取得するには、ICE か LDAP 検索を使用します。これらのユーティリティは rootDSE(ディレクトリサービスエージェント、固有エントリ)から情報を要求します。

RootDSE は、ディレクトリツリーの擬似オブジェクトです。このオブジェクトは、ツリーのルートにある名前のないエントリです。RootDSE は接続しているサーバに固有の情報を持っています。たとえば、rootDSE はスキーマと、スキーマがサポートする拡張およびコントロールの場所の情報を持っています。

rootDSE はツリー内の名前のないエントリであるため、通常、LDAP サーバは検索操作で rootDSE をクライアントに返しませんが、

次の表は、rootDSE から得られる情報を表示したものです。

情報と説明	引用
スキーマの場所:LDAPサーバまたはツリーのスキーマの場所は、subschemaSubentry を読み込むことによって検索できます。eDircotry では、cn=schema が検索のベースになります。	subschemaSubentry:cn=schema

情報と説明	引用
サポートされている拡張：拡張により、コンテンツの作成、マージ、新しいレプリカの追加、LDAP サーバのリフレッシュ、レプリカの削除、レプリカタイプのマスターから読み取り / 書き込みまたは読み取り専用への変更などのサーバの管理、および識別情報の管理ができます。	supportedExtension:2.16.840.1.113719.1.27.100.12 supportedExtension:2.16.840.1.113719.1.27.100.7 supportedExtension: 2.16.840.1.113719.1.27.100.8
拡張の形式は ASN.1OID です。拡張の詳細については、『LDAP Extensions(LDAP 拡張) ( <a href="http://developer.novell.com/ndk/doc/ldapover/ldap_enu/data/a6ik7oi.html">http://developer.novell.com/ndk/doc/ldapover/ldap_enu/data/a6ik7oi.html</a> )』を参照してください。	
LDAP サーバを提供しているベンダ	vendorName:Novell, Inc.
LDAP サーバがサポートしているディレクトリバージョン	vendorVersion:eDirectory v8.7.0 (10410.29)
eDirectory が実行しているバージョン	vendorVersion:eDirectory v8.7.0 (10410.29)
ディレクトリサーバ名とディレクトリツリー名	dsaName:cn=WestWindNDS,o=westwind directoryTreeName:t=WESTWINDTREE
サポートされている SASL メカニズム	supported SASLMechanisms:EXTERNAL supported SASLMechanisms:DIGEST-MD5 supported SASLMechanisms:NMAS LOGIN
サポートされている LDAP サーバのバージョン	supportedLDAPVersion:2 supportedLDAPVersion: 3
サーバ統計情報 :RootDSE は LDAP サーバに関するさまざまな統計情報を提供します ( 強力な認証バインド数など )。	エラー :0 securityErrors:0 chainings:3 referralsReturned:6 extendedOps:0 abandonOps:0 wholeSubtreeSearchOps: 1

rootDSE の情報は、アプリケーション開発に活用することができます。

**シナリオ：アプリケーションを開発する** — あるユーザが新しいレプリカを作成するアプリケーションを作成しています。rootDSE を読み込むと、リストに supportedExtension: 2.16.840.1.113719.1.27.100.7 と記述されています。これにより、サーバが新しいレプリカを作成するコールをサポートすることがわかります。

また、Novell iManager は rootDSE で利用できる機能をチェックし、その情報に従って動作します。

rootDSE を検索するには、ワークステーションで次のように入力します。

```
ldapsearch -h hostname -p 389 -b "" -s base "objectclass=*"
```

この検索は、ldap\_search API を使用したどのアプリケーションでも実行することができます。

検索のベースは NULL で、フィルタは objectclass=\* に設定されています。(このクライアントの場合、ベースは -b です)。



rootDSE の読み取り方法の詳細については、次のいずれかを参照してください。

- ◆ 『LDAP Libraries for C (<http://developer.novell.com/ndk/doc/cldap/ldaplibc/data/hevgtl7k.html>)』
- ◆ 『LDAP Classes for Java (<http://developer.novell.com/ndk/doc/jldap/jldapenu/data/hevgtl7k.html>)』

LDAP 検索フィルタの詳細については、「LDAP Search Filters(LDAP 検索フィルタ) ([http://developer.novell.com/ndk/doc/ldapover/ldap\\_enu/data/a3saoeg.html](http://developer.novell.com/ndk/doc/ldapover/ldap_enu/data/a3saoeg.html))」を参照してください。このセクションは、NDK マニュアルの「LDAP and NDS Integration(LDAP と NDS の統合)」セクションにあります。



# 13

## Novell eDirectory の SNMP サポート

SNMP(Simple Network Management Protocol) は、インターネットを介してデバイスを操作および保守するための標準的なプロトコルです。管理コンソールアプリケーションと管理対象デバイスは、このプロトコルに従って管理情報をやり取りします。管理コンソールアプリケーションには、HP\* Openview、Novell® NMS、IBM\* NetView、Sun\* Net Manager などがあります。管理対象デバイスとしては、ホスト、ルータ、ブリッジ、ハブなどのほか、Novell eDirectory™ のようなネットワークアプリケーションもあります。

このセクションでは、Novell eDirectory 8.7.3 の SNMP サービスについて説明します。このセクションには、次のトピックが含まれています。

- ◆ [335 ページの「SNMP に関する用語の定義」](#)
- ◆ [336 ページの「SNMP サービスについて」](#)
- ◆ [338 ページの「eDirectory と SNMP」](#)
- ◆ [341 ページの「SNMP Services for eDirectory のインストールと設定」](#)
- ◆ [351 ページの「SNMP による eDirectory の監視」](#)
- ◆ [380 ページの「トラブルシューティング」](#)

### SNMP に関する用語の定義

この章で使われる用語の定義を次の表に示します。

用語	定義
EMANATE	SNMP Research International, Inc. の製品。Enhanced MANAGEMENT Agent Through Extensions の略称です。
SNMP	Simple Network Management Protocol の略。ネットワークの稼動状況に関するデータをやり取りするためのプロトコルです。
NAA	Native Agent Adapter の略。ネイティブエージェントアダプタ。
NMS	ネットワーク管理ステーション
MA	Management Agent の略。管理エージェント。
SA	サブエージェント
MIB	管理情報ベース (MIB)
NCP™	Netware® Core Protocol™ の略。
NMA	Network Management Application の略。ネットワーク管理アプリケーション。

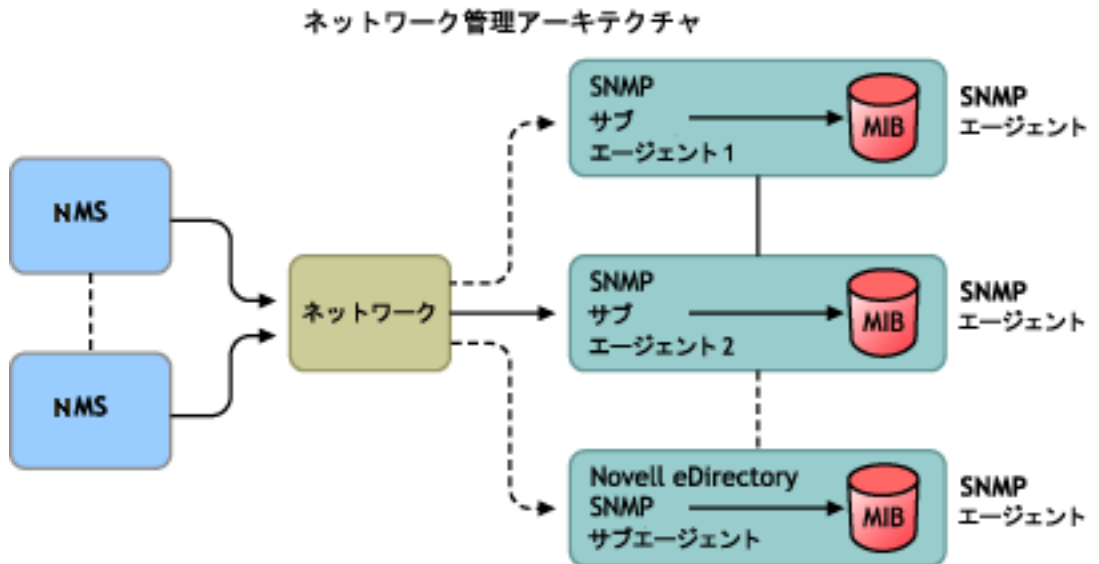
用語	定義
edir.mib	Novell eDirectory サーバの監視に使う MIB のこと。Novell eDirectory に関する MIB オブジェクトおよびトラップが設定されています。
トラップ	eDirectory イベントがサーバ上で発生したときに、管理対象デバイス上のエージェントが発する警告のこと。警告を発する条件は、Novell が提供する MIB(Management Information Base) で定義されています。

## SNMP サービスについて

SNMP は「マネージャ/エージェント」アーキテクチャにもとづくプロトコルです。SNMP を使用して行われるネットワーク管理のアーキテクチャは、次の要素から成ります。

- ◆ NMS(Network Management Station)
- ◆ 管理対象デバイス
- ◆ マスタエージェント
- ◆ サブエージェント
- ◆ MIB(Management Information Base)
- ◆ ネットワーク管理プロトコル

図 33 ネットワーク管理アーキテクチャ



### ネットワーク管理ステーション

ネットワーク管理ステーション (NMS) とは、ネットワーク管理アプリケーションがインストールされたワークステーションのことです。管理対象デバイスに関する情報をグラフィック表示します。

NMS には次の機能があります。

- ◆ ネットワーク管理システム全体のユーザインタフェースを提供します。この機能により、ネットワーク管理の強固さ、柔軟性、および使いやすさが決まります。

- ◆ SNMP Get、Get Next、SNMP GetResponse、および Set の操作はここから実行します。また、ネットワーク上の管理対象デバイスから送られてくる SNMP トラップを捕捉するのも、やはり NMS の役割です。
- ◆ 複数のネットワーク管理アプリケーション (NMA) を同時に監視し、管理対象デバイスに関する情報をグラフィック表示します。表形式で表示する機能、ログとして記録する機能もあります。
- ◆ NMS に組み込まれている MIB コンパイラで、MIB ファイルをコンパイルすることができます。

## 管理対象デバイス

SNMP がインストールされているデバイスは、すべて管理対象デバイスとして扱うことができます。ホスト、ルータ、ブリッジ、ハブなどが管理対象デバイスになります。NMS は管理対象デバイスを監視し、またデバイスと通信します。

NMS と管理対象デバイスとの間では、サブエージェントおよびマスタエージェントという 2 種類のエージェントを介して情報をやり取りします。

## サブエージェント

サブエージェントには、管理対象デバイスに関する情報を集め、マスタエージェントに渡す役割があります。

## マスタエージェント

マスタエージェントには、さまざまなサブエージェントと NMS の間で情報を交換する役割があります。マスタエージェントは、通信相手のサブエージェントと同じホスト上で動作します。

## 管理情報ベース (MIB)

SNMP では、プロトコルデータ単位 (PDU: Protocol Data Unit) という形でネットワーク情報を交換します。PDU には、管理対象デバイスに保存されている変数に関する情報が含まれています。この変数のことを管理オブジェクトと言い、その値とオブジェクト名が NMS に渡されます。管理オブジェクトはすべて管理情報ベース (MIB) に定義されています。MIB はツリー状の階層構造で表される仮想データベースです。

## SNMP のネットワーク管理プロトコル

SNMP の基本関数を次の表に示します。

関数	説明
Get	マネージャがエージェントに情報を要求するために使用します。
Get Next	配列や表から情報を取得する際にマネージャが使用します。
Get Response	マネージャから問い合わせを受けたエージェントが、それに応答するために使用します。
Set	エージェント側の MIB にある変数の値を変更するために、マネージャが使用します。
Trap	あるイベントが発生した際、エージェントがマネージャに通知するために使用します。

SNMP の詳細については、次の Web サイトを参照してください。

- ◆ [NET-SNMP Home Page \(http://net-snmp.sourceforge.net\)](http://net-snmp.sourceforge.net)
- ◆ [SNMP FAQ \(http://www.faqs.org/faqs/snmp-faq/part1\)](http://www.faqs.org/faqs/snmp-faq/part1)
- ◆ [RFC 1157 \(http://www.ietf.org/rfc/rfc1157.txt\)](http://www.ietf.org/rfc/rfc1157.txt)
- ◆ [SNMPLink \(http://www.snmplink.org\)](http://www.snmplink.org)
- ◆ [SNMPInfo \(http://www.snmpinfo.com\)](http://www.snmpinfo.com)
- ◆ [SNMP RFC Standard MIBs and Informative Links \(http://www.wtcs.org/snmp4tpc/snmp\\_rfc.htm\)](http://www.wtcs.org/snmp4tpc/snmp_rfc.htm)
- ◆ [RFC 2605 \(http://ietf.org/rfc/rfc2605.txt?number=2605\)](http://ietf.org/rfc/rfc2605.txt?number=2605)

## eDirectory と SNMP

eDirectory には、ユーザ、アプリケーション、ネットワークデバイス、データなど、数多くのオブジェクトを格納し、管理することができます。eDirectory で管理するオブジェクトが増えてくると、オブジェクトの追加や変更に追従する必要性も増してきます。この問題を解決する手段として SNMP を使用することで、ユーザは eDirectory サーバを監視し、変化に追従できるようになります。

### eDirectory の管理に SNMP を使う利点

- ◆ eDirectory サーバをリアルタイムに監視
- ◆ サードパーティ製 SNMP MIB ブラウザから eDirectory を監視
- ◆ eDirectory が正常に稼動しているか、状況を追跡可能
- ◆ 起こりうる問題を検知時に特定し、対処が可能
- ◆ トラップや統計に関する設定により、対象を選択して監視可能
- ◆ eDirectory に対するアクセス状況をグラフ表示
- ◆ SNMP で収集した履歴データを格納し、分析可能
- ◆ SNMP の Get リクエスト、GetNext リクエストを使った統計機能にも対応
- ◆ プラットフォームを選ばず SNMP ネイティブマスタエージェントを使用可能

### eDirectory での SNMP の機能について

SNMP を eDirectory に実装すると、アクセス状況、稼動状況、エラー、キャッシュ性能に関する eDirectory の統計情報を取得できます。また、イベントが発生すると SNMP からトラップが送信されます。トラップや統計情報は MIB に定義されています。

#### ディレクトリサービス監視 MIB

eDirectory の MIB には、eDirectory を監視するための統計情報やトラップが定義されています。この MIB には次の oid(オブジェクト ID) が割り当てられています。

iso(1).org(3).dod(6).internet(1).private(4).enterprise(1).novell(23).mibDoc(2).ndsMIB(98)

## 統計情報

eDirectory MIB は、管理対象オブジェクトを次の 4 つの異なるテーブルに分けて格納しています。

- ◆ **キャッシュデータベース統計テーブル - ndsDbCacheTable:** ディレクトリサーバに関する記述と、サーバにキャッシュされたエントリに関する統計情報の要約を格納します。
- ◆ **設定データベース統計テーブル - ndsDbConfigTable:** ディレクトリサーバに関する記述と、サーバで設定されたエントリに関する統計情報の要約を格納します。
- ◆ **プロトコル統計テーブル - ndsProtoIfOpsTable:** ディレクトリサーバのアプリケーションプロトコルインタフェースごとに、アクセス状況、稼動状況、エラーに関する統計情報の要約を格納します。
- ◆ **相互通信統計テーブル - ndsServerIntTable:** 監視対象ディレクトリが通信した、あるいは通信を試みたディレクトリサーバを、最新の「N」回分記録しておきます。「N」はローカルで定義する定数です。

注: 統計情報の詳細については、[375 ページの「統計情報」](#)を参照してください。

## トラップ - ndsTrapVariables

eDirectory MIBには119種類のトラップが定義されています。そのうち117種類はeDirectoryのイベントにマップされています。ほかに `ndsServerStart` および `ndsServerStop` というトラップがあり、これは SNMP サブエージェントが直接生成します。

注: トラップの詳細については、[351 ページの「トラップ」](#)を参照してください。

統計情報やトラップの詳細については、`edir.mib` も参照してください。

`edir.mib` は次のディレクトリにあります。

```
NetWare:sys:\etc
Windows:<install directory>\SNMP
UNIX:/etc/ndssnmp/
```

## SNMP グループオブジェクト

SNMP グループオブジェクトは、eDirectory SNMP トラップの設定や管理に使います。インストールの過程で、「SNMP Group - *server\_name*」という名前の SNMP グループオブジェクトが作られます (ここで *server\_name* は、eDirectory の SNMP サービスをインストールしたサーバ名を表します)。この SNMP グループオブジェクトが作られるのは、サーバオブジェクトと同じコンテナ内です。SNMP 設定ユーティリティは、SNMP トラップの設定に使います。

### Windows の場合

SNMP グループオブジェクトを作るには、次のコマンドを実行してください。

```
rundll32 snmpinst, snmpinst -c <createobj> -a <userFDN> -p
<password> -h <hostname or IP address>
```

パラメータ	説明
-c <createobj>	オブジェクトの作成を示すトラップコマンド。
-a <userFDN>	管理者権を持つユーザの完全識別名
-p <password>	認証に使う userFDN パスワード
-h <hostname or IP address>	DNS ホスト名または IP アドレス。

例：

```
rundll32 snmpinst, snmpinst -c createobj -a admin.mycontext -p mypassword -h 160.98.146.26
```

SNMP グループオブジェクトを削除するには、次のコマンドを実行してください。

```
rundll32 snmpinst, snmpinst -c <deleteobj> -a <userFDN> -p  
<password> -h <hostname or IP address>
```

詳細については、上の表を参照してください。

例：

```
rundll32 snmpinst, snmpinst -c deleteobj -a admin.mycontext -p mypassword -h 160.98.146.26
```

### NetWare の場合

SNMP グループオブジェクトの作成や削除には `snmpinst` というユーティリティを使います。これは `sys:\system\directory` にあります。

SNMP グループオブジェクトを作るには、次のコマンドを実行してください。

```
SNMPINST -c <adminContext> <password> <ServerDN>
```

パラメータ	説明
-c	オブジェクトの作成を示すフラグコマンド。削除する場合は「-d」と指定します。
<adminContext>	管理者権を持つユーザの完全識別名
<password>	認証に使う userFDN パスワード
<ServerDN>	DNS ホスト名

例：

```
SNMPINST -c admin.mycontext.treename mypassword myserver
```

SNMP グループオブジェクトを削除するには、次のコマンドを実行してください。

```
SNMPINST -d <adminContext> <password> <ServerDN>
```

パラメータについてはオブジェクトを作る場合と同様です。

例：

```
SNMPINST -d admin.mycontext.treename mypassword myserver
```

### UNIX の場合

SNMP グループオブジェクトを作るには、次のコマンドを実行してください。

```
ndsconfig add -m <modulename> -a <userFDN>
```

例：

```
ndsconfig add -m snmp -a admin.mycontext
```



# SNMP Services for eDirectory のインストールと設定

SNMP service for eDirectory は、eDirectory をインストールする際に組み込まれます。SNMP services for eDirectory のデフォルト設定の変更には iManager を使います。詳細については、[343 ページの「ダイナミック設定」](#)を参照してください。

SNMP Group-Object という名前の新しいオブジェクトが、eDirectory のインストール時に、ディレクトリツリーに追加されます。このオブジェクトは Novell eDirectory SNMP トラップの設定や管理に使います。詳細については、[339 ページの「SNMP グループオブジェクト」](#)を参照してください。

## eDirectory をインストール済みの Windows に SNMP を組み込む手順

eDirectory のインストール時に SNMP サービスを除外した場合は、SNMP サブエージェント用ファイルがコピーされるだけで、レジストリは元のままになっています。

あとになって SNMP サービスが必要になった場合は、次のコマンドでレジストリを更新してください。

```
rundll32 snmpinst, snmpinst -c createreg
```

## SNMP サーバモジュールのロードとアンロード

SNMP サーバモジュールは手動でロード、アンロードできます。デフォルトでは、すべてのプラットフォームで自動的にロードされます。Windows および UNIX プラットフォームの場合は、手動でロードすることも可能です。

SNMP サーバモジュールをロードするには、次のコマンドを入力します。

サーバ	コマンド
NetWare	手動でのアンロードは不可
Windows	[DHOST (NDSCONS)] 画面で Ndsnmp.dlm を選択し、[開始] をクリックします。
Linux、Solaris、AIX、HP-UX	DHOST の [リモート管理] ページで、[SNMP Trap Server for Novell eDirectory 8.7.3] アイコンをクリックすると、SNMP トラップサーバがロードされます。  または  コマンドプロンプトから「/usr/bin/ndssnmp -1」と入力する方法もあります。

SNMP サーバモジュールをアンロードするには、次のコマンドを入力します。

サーバ	コマンド
NetWare	手動でのアンロードは不可
Windows	[DHOST (NDSCONS)] 画面で ndssnmp.dlm を選択し、[停止] をクリックします。

サーバ	コマンド
Linux、Solaris、AIX、HP-UX	DHOST の[リモート管理]ページで、[SNMP Trap Server for Novell eDirectory 8.7.3] アイコンをクリックすると、SNMP トラップサーバがアンロードされます。  または  コマンドプロンプトから「 <code>/usr/bin/ndssnmp -u</code> 」と入力する方法もあります。

## サブエージェントの設定

### スタティック設定

スタティック設定は、サブエージェントを起動する前に行います。手動で設定する場合、Windows、Solaris、Linux、AIX の場合は `ndssnmp.cfg`、NetWare の場合は `dssnmp.cfg` というファイルを編集してください。ファイル `ndssnmp.cfg` は次のディレクトリにあります。

Windows:`install_directory\SNMP\`

NetWare:`sys\etc\`

UNIX:`/etc/ndssnmp/`

注: `ndssnmp.cfg` を書き替えた場合、サブエージェントを再起動する必要があります。

サブエージェントの設定は、次の書式で記述してください。

- ◆ **INTERACTIVE *status***

ここで *status* には、「on」または「off」を指定します。「on」を指定すると、サブエージェント起動時にユーザ名とパスワードを要求されます。ステータスが Off の場合、ユーザ名およびパスワードはセキュリティ保護された保存データから取得されません。デフォルト値は「off」です。

例:

INTERACTIVE on

INTERACTIVE off

- ◆ **INTERACTION *value***

ここで *value* は、インタラクションテーブルのエントリ数を表します。値の範囲は 1 ~ 10 で、デフォルト値は 4 です。例: INTERACTION 4 INTERACTION 2

- ◆ **MONITOR *status***

ここで *status* には、「on」または「off」を指定します。デフォルト値は「on」です。

例:

MONITOR on

MONITOR off

- ◆ **SSLKEY *certificate\_file***

ここで *certificate\_file* は、証明書のエクスポート先パスを表します。エクスポートした証明書が実際に存在するパスを指定してください。

例:

SSLKEY /home/guest/snmp-cert.der (UNIX の場合)

SSLKEY c:\home\guest\snmp-cert.der (Windows NT、NetWare の場合)

- ◆ `SERVER hostname/ipaddr`

ここで `hostname` は、eDirectory サーバをインストールし、設定したホスト名を表します。指定できるのは、ローカルにインストールしたサーバに限ります。

この指定は必須です。指定がない場合はどのサーバも監視の対象になりません。デフォルト：ローカルサーバのホスト名。

例：

```
SERVER myserver
```

```
SERVER myserver:1524
```

注：このコマンドで、コロン(:)の前にはスペースを入れないでください。

## ダイナミック設定

ダイナミック設定は、ディレクトリサービスの稼動中、いつでも次のいずれかの方法で実行できます。

### コマンドラインからの設定

トラップ設定用のコマンドラインユーティリティを使用して、eDirectory の SNMP トラップを設定できます。


次のような操作が可能です。

- ◆ トラップの有効化/無効化、トラップ間隔の設定、エラートラップの有効化/無効化、有効な/無効な/すべてのトラップのリスト表示

注：詳細については、[366 ページの「トラップに関する設定」](#)を参照してください。

### iManager プラグインによる設定

トラップの設定には、Novell iManager を使う方法もあります。Novell iManager はブラウザベースのツールで、eDirectory オブジェクトを運用、管理、設定するために使用します。iManager を使用すると、ユーザに特定のタスクや責任を割り当てたり、それらのタスクを実行するために必要なツールおよびそれに伴う権利だけを付与したりすることができます。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [SNMP Management] > [SNMP の概要] の順にクリックします。
- 3 [SNMP グループオブジェクトを表示] をクリックして、設定する SNMP グループオブジェクトの名前をクリックします。
- 4 [一般] ページまたは [トラップ] ページで、必要なパラメータを設定します。
- 5 [適用] をクリックし、[OK] をクリックすると、新しい設定内容が保存されます。

注：詳細については Novell iManager のオンライン・ヘルプを参照してください。

## SNMP Services for eDirectory の設定

SNMP Services for eDirectory の設定は、次の手順で行います。

1. マスタエージェントの設定
2. マスタエージェントの起動
3. サブエージェントの設定
4. サブエージェントの起動

## NetWare の場合

NetWare の場合、ネイティブマスタエージェント (snmp.nlm) が、初めからオペレーティングシステムに組み込まれています。

ヒント : NetWare にはデフォルトの SNMP マスタエージェントがあります。詳細については、[SNMP Developers Components \(http://developer.novell.com/ndk/snmpcomp.htm\)](http://developer.novell.com/ndk/snmpcomp.htm) を参照してください。

### マスタエージェントの設定

コミュニティ名

- 1 コマンドプロンプトから、「**inetcfg**」と入力します。
- 2 [Manage Configuration] オプションを選択します。
- 3 [Configure SNMP parameters] オプションを選択します。
- 4 コミュニティ名を表す文字列を適宜編集してください。

トラップの送信先

- 1 ファイル `sys:\etc\traptarg.cfg` を編集し、トラップの送信先とするコンピュータの IP アドレスまたはホスト名を指定します。

### マスタエージェントの起動

マスタエージェント snmp.nlm はデフォルトで起動されています。

### サブエージェントのロード

- 1 サブエージェントをロードするには、コマンドプロンプトから「**dssnmpsa**」と入力します。  
ダイアログボックスが表示されます。[ログイン] オプション、[終了] オプションがあります。
- 2 処理を続行する場合は [ログイン]、中断する場合は [終了] を選択します。
- 3 (状況によって実行) [ログイン] を選択した場合、ユーザ名とパスワードを求められるので入力してください。
- 4 [パスワードを保存する] に「**y**」と入力すると、パスワードを保存できます。次回は、パスワードを入力することなくサブエージェントを起動できます。「**N**」と入力した場合は、次回もパスワードを求められます。
- 5 いずれかを入力した後、<Enter> キーを押してください。
- 6 <F10> キーを押すとツリーにログインできます。
- 7 <Enter> キーを押して操作を続けます。
- 8 サブエージェントが起動されます。

注: このダイアログボックスは、ファイル `sys:\etc\ndssnmp.cfg` で INTERACTION が「on」に設定されている場合に表示されます。「off」の場合は表示されません。

## Windows の場合

### マスタエージェントの設定

注: SNMP マスタエージェントは、eDirectory をインストールする前に組み込んでおく必要があります。詳細については、『[SNMP Installation on Windows\(SNMP の Windows へのインストール\)](http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnolog/winntas/maintain/featusability/getting.asp) (http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnolog/winntas/maintain/featusability/getting.asp)』を参照してください。

- 1 [Microsoft SNMP Properties] ダイアログボックスを開き、[エージェント] タブをクリックします。
- 2 接続先および場所に関する情報を入力します。
- 3 [トラップ] タブをクリックし、コミュニティ名およびトラップの送信先に関する情報を入力します。
  - 3a コミュニティ名を入力し、[追加] をクリックします。
  - 3b トラップの送信先のコンピュータの IP アドレスまたはホスト名を入力してください。
  - 3c [追加] をクリックすると、ここで入力した IP アドレスまたはホスト名が追加されます。
- 4 [デスクトップとの対話をサービスに許可する] オプションを有効にします。

このオプションを有効にしない場合、Windows 上の SNMP には接続できません。

  - ◆ Windows NT の場合：[スタート] > [設定] > [コントロールパネル] > [サービス] の順にクリックします。次に [SNMP] > [起動] をクリックし、[デスクトップとの対話をサービスに許可する] を選択します。
  - ◆ Windows 2000 の場合：[スタート] > [設定] > [コントロールパネル] > [管理ツール] > [サービス] の順にクリックします。[SNMP] を右クリックし、[プロパティ] を選択します。[ログオン] タブで、[デスクトップとの対話をサービスに許可する] を選択します。

### マスタエージェントの起動

マスタエージェントを起動するには、次のいずれかの操作を実行してください。

- ◆ Windows NT の場合：[スタート] > [設定] > [コントロールパネル] > [サービス] > [SNMP] > [開始] の順にクリックします。

Windows 2000 の場合：[スタート] > [設定] > [コントロールパネル] > [管理ツール] > [サービス] > [SNMP] > [開始] の順にクリックします。
- ◆ コマンドラインで次を入力します。

```
Net start SNMP
```

### マスタエージェントの停止

マスタエージェントを停止するには、次のいずれかの操作を実行してください。

- ◆ Windows NT の場合：[スタート] > [設定] > [コントロールパネル] > [サービス] > [SNMP] > [停止] の順にクリックします。

Windows 2000 の場合：[スタート] > [設定] > [コントロールパネル] > [管理ツール] > [サービス] > [SNMP] > [停止] の順にクリックします。
- ◆ コマンドラインで次を入力します。

```
Net stop SNMP
```

### サブエージェントの起動

Windows の場合、マスタエージェントを起動すると、サブエージェントも自動的に起動されます。

**重要：**最新の Service Pack は、SNMP サービスのインストール後にインストールする必要があります。

### マスタエージェントの設定

SNMP パッケージをロードする前に、Solstice Enterprise master agent 1.0.3 をインストールしておく必要があります。まだインストールされていない場合は、[Solstice Enterprise Agents \(http://www.sun.com/software/entagents\)](http://www.sun.com/software/entagents) Web サイトからダウンロードしてください。

- 1 ファイル `/etc/snmp/conf/snmpd.conf` でホスト名を設定します。次のトラップエントリを追加します。

```
trap myserver
```

ここで `myserver` は、トラップの送信先ホスト名を表します。

- 2 ファイル `/etc/snmp/conf/snmpdx.acl` のトラップパラメータ部に、次の設定を追加します。

```
trap-community = public
hosts = myserver {
enterprise = "Novell eDirectory"
trap-num = 1-119 }
```

ここで、`trap-community` はトラップに使うコミュニティ名、`myserver` はトラップの送信先ホスト名、`Novell eDirectory` はエンタープライズ MIB、`trap-num` はトラップの範囲を表します。

**重要:** 設定ファイルを変更した場合、マスタエージェントとサブエージェントを再起動する必要があります。

### マスタエージェントの起動

マスタエージェントを起動するには、次のコマンドを実行してください。

```
/usr/lib/snmp/snmpdx -y -c /etc/snmp/conf
```

### サブエージェントの設定

Solaris では、サブエージェント `ndssnmpsa` はデーモンとして動作します。

サブエージェントの設定には、`/etc/snmp/conf` 以下にある、次の設定ファイルを使います。

- ◆ `ndsmib.reg` はサブエージェントの登録ファイルです。
- ◆ `ndsmib.acl` は SNMP サブエージェントの設定ファイルです。

### サブエージェントの起動

サブエージェントを起動するには、次のコマンドを実行します。

```
/etc/init.d/ndssnmpsa start
```

ユーザ名とパスワードを求められたら入力します。正常に認証されると、`/etc/ndssnmp/ndssnmp.cfg` で `INTERACTION = on` という設定になっている場合は次のメッセージが表示されます。

```
Do you want to remember password?(Y/N)
```

ここで「**Y**」と入力すると、パスワードが保存されます。次回からは、パスワードを入力することなくサブエージェントを起動できます。

「**N**」と入力した場合は、次回もパスワードを求められます。

### サブエージェントの停止

サブエージェントを停止するには、次のコマンドを実行してください。

```
/etc/init.d/ndssnmpsa stop
```

### マスタエージェントの設定

Linux の場合、ucd-snmp-4.2.1-7.rpm 以降をインストールしておく必要があります。

注: ucd-snmp パッケージは net-snmp という名前に変わりました。詳細については、<http://www.net-snmp.org/> (<http://www.net-snmp.org/>) を参照してください。

ファイル /etc/snmp/snmpd.conf でホスト名を設定します。

```
trapsink myserver public
```

ここで *myserver* は、トラップの送信先ホスト名を表します。

ファイル /etc/snmp/snmpd.conf に、次の記述を追加します。

```
master agentx
```

さらに、次の設定も変更する必要があります。

元のコンテンツ	変更後のコンテンツ
com2sec notConfigUser default public	com2sec demouser default public
group notConfigGroup v1 notConfigUser	group demogroup v1 demouser
view systemview included system	view all included .1
access notConfigGroup "" any noauth exact systemview none none	access demogroup "" any noauth exact all all all

**重要:** 設定ファイルを書き替えた場合、マスタエージェントとサブエージェントを再起動する必要があります。

### マスタエージェントの起動

マスタエージェントを起動するには、次のコマンドを実行します。

```
/usr/sbin/snmpd
```

### サブエージェントの起動

サブエージェントを起動するには、次のコマンドを実行します。

```
/etc/rc.d/init.d/ndssnmpsa start
```

ユーザ名とパスワードを求められたら入力します。正常に認証されると、/etc/ndssnmp/ndssnmp.cfg で INTERACTION = on という設定になっている場合は次のメッセージが表示されます。

```
Do you want to remember password?(Y/N)
```

ここで「**Y**」と入力すると、パスワードが保存されます。次回は、パスワードを入力することなくサブエージェントを起動できます。

「**N**」と入力した場合は、次回もパスワードを求められます。

### サブエージェントの停止

サブエージェントを停止するには、次のコマンドを実行してください。

```
/etc/rc.d/init.d/ndssnmpsa stop
```

### マスタエージェントの設定

ファイル `/etc/snmpd.conf` で、トラップの送信先を次のように記述してください。

```
trap community myserver view_name trap_mask
```

ここで、

- ◆ `community` はトラップパケットにエンコードされるコミュニティ名を表します。
- ◆ `myserver` は、トラップの送信先ホスト名を表します。
- ◆ `view_name` は一意的なオブジェクト識別子で、数値をドット (.) で区切った書式で表します。

次に例を示します。1.3.6.1.4.1.23.2.98。このパラメータの指定は任意です。指定がない場合は、デフォルトにより MIB ツリー全体が表示されます。

- ◆ `trap_mask` は 16 進数で表します。

上位ビットから順に、`coldStart`、`warmStart`、`linkDown`、`linkUp`、`authenticationFailure`、`egpNeighborLoss`、`enterpriseSpecific` の各トラップに対応します。例の中で、一番右の「98」という値には何の意味もありません。「1」になっているビットに対応したトラップが送信されます。それ以外のトラップはブロックされます。

例：

```
fe  どのトラップもブロックしません (1111 1110)
```

```
7e  coldStart トラップをブロックします (0111 1110)
```

```
be  warmStart トラップをブロックします (1011 1110)
```

```
3e  coldStart トラップ、warmStart トラップをブロックします (0011 1110)
```

### マスタエージェントの起動

マスタエージェントを起動するには、次のコマンドを実行します。

```
/usr/sbin/snmpd
```

### サブエージェントの起動

サブエージェントを起動するには、次のコマンドを実行します。

```
/etc/ndssnmpsa start
```

ユーザ名とパスワードを求められたら入力します。正常に認証されると、`/etc/ndssnmp/ndssnmp.cfg` で `INTERACTION = on` という設定になっている場合は次のメッセージが表示されます。

```
Do you want to remember password?(Y/N)
```

ここで「**Y**」と入力すると、パスワードが保存されます。次回は、パスワードを入力することなくサブエージェントを起動できます。

「**N**」と入力した場合は、次回もパスワードを求められます。

### サブエージェントの停止

サブエージェントを停止するには、次のコマンドを実行してください。

```
/etc/ndssnmpsa stop
```



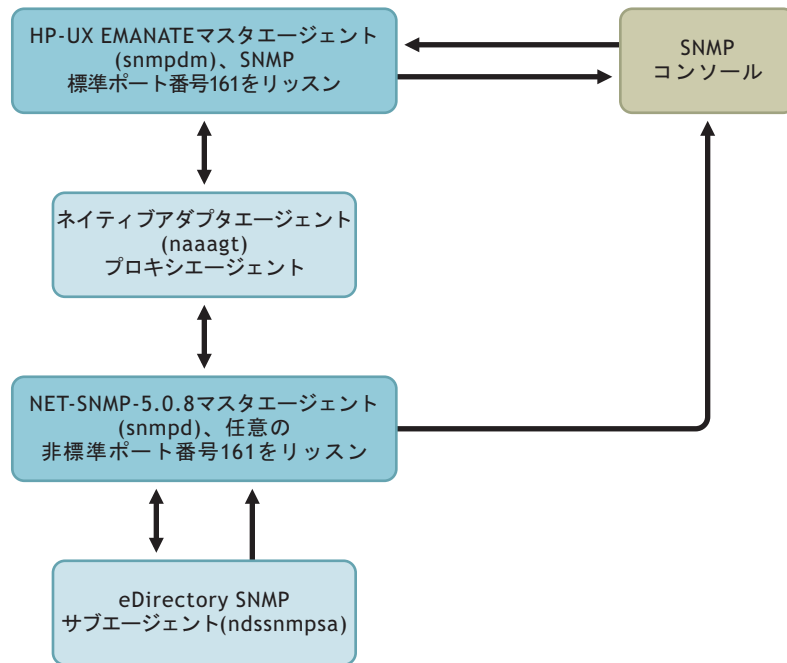
## HP-UX の場合

HP-UX の場合、EMANATE SNMP というネイティブマスターエージェントがあります。マスターエージェントを設定する場合、プロキシ SNMP エージェントの設定も必要です。プロキシエージェントの設定には NAA(Native Adapter Agenet) を使います。NAA には、サードパーティ製 SNMP エージェントが、HP-UX SNMP マスターエージェント (snmpdm) と協調して動作できるようにする働きがあります。ここでは NET-SNMP マスターエージェントが、サードパーティ製 SNMP エージェントに相当します。NET-SNMP マスターエージェントは、NAA の設定と同じ非標準 UDP ポートをリッスンする必要があります。

詳しくは、350 ページの「ネイティブエージェントアダプタ (NAA) の起動と設定」および 350 ページの「NET-SNMP マスターエージェントの起動 / 設定」を参照してください。

eDirectory SNMP サブエージェント、NET-SNMP マスターエージェント、NAA エージェント、HP-UX EMANATE マスターエージェント、SNMP コンソール間のデータの流れを図に示します。

図 34 SNMP データの流れ



### HP-UX SNMP マスターエージェントの起動

HP-UX SNMP マスターエージェントを起動するには、次のコマンドを実行します。

```
/etc/snmpd
```

または

```
/usr/sbin/snmpdm
```

注：HP-UX SNMP マスターエージェントを停止するには、「/etc/snmpd -k」と入力します。

## ネイティブエージェントアダプタ (NAA) の起動と設定

NAA エージェント (naaagt) を起動する前に、次の環境変数をエクスポートしておく必要があります。

- ◆ HP\_NAA\_CNF - NAA 設定ファイル名
- ◆ HP\_NAA\_PORT - NET-SNMP マスタエージェントがリッスンする非標準 UDP ポート番号
- ◆ HP\_NAA\_GET\_COMMUNITY - NAA から NET-SNMP マスタエージェントへの SNMP 要求で使うコミュニティ名。

次に例を示します。

```
export HP_NAA_CNF=/etc/ndssnmp/ndssnmpNAA.cfg
export HP_NAA_PORT=8161 ## Specify any non-standard UDP port
export HP_NAA_GET_COMMUNITY=public
```

NAA エージェントについて詳しくは、naaagt のマニュアルページを参照してください。

次のコマンドを実行すると、NAA エージェントが起動されます。

```
/usr/sbin/naaagt
```

注: NAA エージェントを起動するには root としてのアクセス権が必要です。

## NET-SNMP マスタエージェントの起動 / 設定

NET-SNMP マスタエージェントを設定する前に、まずこれをダウンロードし、インストールしておく必要があります。

- 1 NET-SNMP バージョン 5.0.8 tar ファイル (net-snmpp-5.0.8-HP-UX\_B.11.00\_9000\_712.tar.gz) を、SourceForge.net ([http://sourceforge.net/project/showfiles.php?group\\_id=12694](http://sourceforge.net/project/showfiles.php?group_id=12694)) からダウンロードします。
- 2 NET-SNMP バージョン 5.0.8 のバイナリをインストールするには、上記 tar ファイルを展開します。

tar ファイルを展開すると、NET-SNMP バージョン 5.0.8 のバイナリが *current\_working\_directory*/usr/local にインストールされます。

NET-SNMP マスタエージェントを、次の手順で設定します。

- ◆ ファイル /etc/ndssnmp/snmpd-net-snmpp.conf でホスト名を設定します。  
trapsink *myserver* public  
ここで *myserver* は、トラップの送信先ホスト名を表します。
- ◆ /etc/ndssnmp/snmpd-net-snmpp.conf に次の記述がなければ追加します。  
master agentx

注: NET-SNMP 5.0.8 にはマスタエージェント設定ファイルのサンプルが付属していません。eDirectory SNMP コンポーネントに付属しているサンプルを使用してください。eDirectory をインストールすると、/etc/ndssnmp 以下にサンプルファイル snmpd-net-snmpp.conf が格納されます。

NET-SNMP 5.0.8 マスタエージェントは次のコマンドで起動します。

```
installed_NET-SNMP_directory/usr/local/sbin/snmpd -C -c /etc/ndssnmp/snmpd-net-snmpp.conf 8161
```

重要: 設定ファイルを変更した場合、マスタエージェントとサブエージェントを再起動する必要があります。

## サブエージェントの起動

サブエージェントを起動するには、次のコマンドを実行します。

```
/sbin/init.d/ndssnmpsa start
```

ユーザ名とパスワードを求められたら入力します。正常に認証されると、`/etc/ndssnmp/ndssnmp.cfg` で `INTERACTION = on` という設定になっている場合は次のメッセージが表示されます。

```
Do you want to remember password?(Y/N)
```

ここで「**Y**」と入力すると、パスワードが保存されます。次回は、パスワードを入力することなくサブエージェントを起動できます。

「**N**」と入力した場合は、次回もパスワードを求められます。

## サブエージェントの停止

サブエージェントを停止するには、次のコマンドを実行します。

```
/sbin/init.d/ndssnmpsa stop
```

# SNMP による eDirectory の監視

SNMP のトラップおよび統計機能を使用して、eDirectory の動作を監視できます。

ただし、そのためには、NCP サーバ、LDAP グループ、LDAP サーバオブジェクトに対して、次の権利が必要です。

- ◆ NCP サーバオブジェクトに対するスーパーバイザ権
- ◆ LDAP グループオブジェクトの LDAP Allow Clear Text Password 属性に対する読み込み権
- ◆ LDAP サーバオブジェクトの LDAP TCP Port 属性、LDAP SSL Port 属性に対する読み込み権

デフォルトでは、ユーザが管理者としてログインした場合、SNMP を使用した eDirectory サーバの監視で問題が生じることはありません。

## トラップ

eDirectory にはトラップを生成するイベントが全部で 119 種類あります。トラップの生成状況は MIB ブラウザで確認できます。

注：42 番、92 番、100 番のトラップは、NetWare の場合しか生成されません。

トラップ 番号	トラップ名	生成される条件
1	ndsCreateEntry	新規オブジェクトがディレクトリに追加されたとき。  例：  LDAP ツール、ICE、ConsoleOne®、または iManager などを使用してオブジェクトを生成したとき。

トラップ 番号	トラップ名	生成される条件
2	ndsDeleteEntry	既存のオブジェクトが削除されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用してオブジェクトを削除したとき。
3	ndsRenameEntry	既存のオブジェクト名が変更されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用してオブジェクト名を変更したとき。
4	ndsMoveSourceEntry	オブジェクトのコンテキストが変わったとき。このトラップにより、変更前のコンテキストが通知されます。  例：  ldapmodrdn や ldapsdk でオブジェクトを移動したとき。
5	ndsAddValue	オブジェクトの属性値が追加されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、属性に新しい値を追加したとき。
6	ndsDeleteValue	オブジェクトの属性値が削除されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、属性値を削除したとき。
7	ndsCloseStream	ストリーム属性が変更されたとき。
8	ndsDeleteAttribute	オブジェクトの属性値（単一値と定義されているもの）が削除されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、属性値を削除したとき。
9	ndsCheckSecurityEquiv	あるエントリの同等セキュリティベクトルが検査されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、同等セキュリティ属性を変更したとき。
10	ndsUpdateSecurityEquiv	あるエントリの同等セキュリティベクトルが変更されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、同等セキュリティ属性を変更したとき。

トラップ 番号	トラップ名	生成される条件
11	ndsMoveDestEntry	オブジェクトのコンテキストが変わったとき。このトラップにより、変更後のコンテキストが通知されます。  例：  ldapmodrdn や ldapsdk でオブジェクトを移動したとき。
12	ndsDeleteUnusedExtref	バックリンクオブジェクトが削除されたとき。
13	ndsAgentOpenLocal	ローカルディレクトリエージェントがオープンされたとき。  例：  無人修復を実行したとき。
14	ndsAgentCloseLocal	ローカルディレクトリエージェントがクローズされたとき。  例：  無人修復を実行したとき。
15	ndsDSABadVerb	DSAgent 要求に関連づけられた動詞番号が正しくないとき。  例：  DClient 呼び出しを使用して、eDirectory に不正なバープ要求を送ったとき。
16	ndsMoveSubtree	コンテナがそれに含まれるオブジェクトと共に移動されたとき。  例 :LDAP ツール、ICE、ConsoleOne、または iManager を使用して、パーティションを他のコンテキストに移動したとき。
17	ndsNoReplicaPointer	レプリカにレプリカポインタが関連づけられていないとき。
18	ndsSynclnEnd	インバウンド同期が終了したとき。
19	ndsBacklinkSecurEquiv	バックリンク操作により、オブジェクトの同等セキュリティベクトルが更新されたとき。  例：  LDAP ツール、ICE、ConsoleOne、または iManager を使用して、同等セキュリティ属性を変更したとき。
20	ndsBacklinkOperPrivChg	バックリンク操作により、オブジェクトのコンソールオペレータ特権が変更されたとき。
21	ndsDeleteSubtree	コンテナオがそれに含まれるオブジェクトと共に削除されたとき。
22	ndsReferral	参照が作成されたとき。

トラップ 番号	トラップ名	生成される条件
23	ndsUpdateClassDef	スキーマクラス定義が更新されたとき。  例：  新規クラスまたは属性をプライマリサーバに追加し、LDAP ツール、ICE、ConsoleOne、または iManager を使用してセカンダリサーバと同期を取った場合、このトラップが生成されます。
24	ndsUpdateAttributeDef	スキーマ属性定義が更新されたとき。  例：  新規属性をプライマリサーバに追加し、セカンダリサーバ側で、LDAP ツール、ICE、ConsoleOne、または iManager を使用して同期したとき。
25	ndsLostEntry	eDirectory のローカルサーバには存在しないはずのエントリに対する更新要求があったとき。
26	ndsPurgeEntryFail	ページ処理に失敗したとき。
27	ndsPurgeStart	ページ処理を開始したとき。  例：  dstrace を実行し、ndstrace=*j と設定したとき。
28	ndsPurgeEnd	ページ処理が終了したとき。  例：  dstrace を実行し、ndstrace=*j と設定したとき。
29	ndsLimberDone	リンバ処理が終了したとき。  例：  Dstrace を設定して、所定の時間間隔でリンバを開始したとき。
30	ndsPartitionSplitDone	パーティション分割処理が終了したとき。  例：  ConsoleOne または iManager を使用して、パーティションを作成したとき。
31	ndsSyncServerOutStart	ある特定のサーバに同期するアウトバウンド同期処理が起動されたとき。  例：  dstrace の設定により、所定の時間間隔でアウトバウンド同期処理が起動されたとき。

トラップ 番号	トラップ名	生成される条件
32	ndsSyncServerOutEnd	ある特定のサーバに同期するアウトバウンド同期処理が終了したとき。  例：  dstrace の設定により、所定の時間間隔でアウトバウンド同期処理が起動され、終了したとき。
33	ndsSyncPartitionStart	パーティション同期処理が起動されたとき。  例：  あるコンテナをパーティション分割したとき。
34	ndsSyncPartitionEnd	パーティション同期処理が終了したとき。  例：  あるコンテナをパーティション分割したとき。
35	ndsMoveTreeStart	サブツリーの移動処理が起動されたとき。  パーティションの移動に伴い、サブツリーも移動します。  例：  ConsoleOne または iManager を使用して、パーティションを作成し、他のコンテナに移動したとき。
36	ndsMoveTreeEnd	サブツリーの移動処理が終了したとき。  パーティションの結合処理に伴い、サブツリーも移動します。  例：  ConsoleOne または iManager を使用して、パーティションを作成し、他のコンテナに移動したとき。
37	ndsJoinPartitionDone	パーティションの結合処理が終了したとき。  例：  ConsoleOne または iManager を使用して、パーティションを作成し、他のパーティションと結合したとき。
38	ndsPartitionLocked	パーティションがロックされたとき (結合処理の際など)。  例：  ConsoleOne または iManager を使用して、パーティションを作成したとき。
39	ndsPartitionUnlocked	パーティションのロックが解除されたとき (結合処理が終了したときなど)。  例：  ConsoleOne または iManager を使用して、パーティションを作成したとき。

トラップ 番号	トラップ名	生成される条件
40	ndsSchemaSync	スキーマの同期処理が起こったとき。  例：  ldapsdk schsync のスケジュール設定により、スキーマの同期処理が始まったとき。
41	ndsNameCollision	他のサーバにある別のオブジェクトと名前が重複（衝突）したとき。  例：  iMonitor を使用して、プライマリサーバとセカンダリサーバのアウトバウンド同期処理を無効にし、LDAP ツールを使用して両サーバにユーザオブジェクトを追加します。その後、iMonitor でアウトバウンド同期処理を実行した場合、衝突が起こります。
42	ndsNLMLoaded	NetWare で、NLM™ プログラムがロードされたとき。  このトラップは NetWare でのみ発生します。  例：  nldap.nlm をロードまたはアンロードしたとき。
43	ndsChangeModuleState	eDirectory モジュール(NLM/DLM)がロードまたはアンロードされたとき。  例：  nldap モジュールをロードまたはアンロードしたとき。
44	ndsLumberDone	バックグラウンドでのリンバ処理が始まったとき。
45	ndsBacklinkProcDone	バックリンク処理が終了したとき。  例：  dstrace の設定により、所定の時間間隔でバックリンク処理が起動され、終了したとき。
46	ndsServerRename	サーバ名が変更されたとき。  例：  ldapmodrdn または ldapsdk を使用して、サーバ名を変更したとき。
47	ndsSyntheticTime	将来のタイムスタンプを付与したオブジェクトが作成されたとき。eDirectory サーバの同期に際しては、合成時刻として扱うことになります。  例：  ndsconfig を使用して、セカンダリサーバをツリーに追加したとき。



トラップ 番号	トラップ名	生成される条件
48	ndsServerAddressChange	リンバによりサーバ参照が変更されたとき。  例：  サーバの IP アドレスを変更して ndsd を再起動したとき。
49	ndsDSARead	エントリが読み出されたとき。  eDirectory を対象とする操作があれば、必ずこのトラップが生成されます。  例：  ldapsearch を実行したとき。
50	ndsLogin	eDirectory へのログインがあったとき。  例：  ndslogin を使用してツリーにログインしたとき。
51	ndsChangePassword	パスワードが変更されたとき。  例：  ldapmodify を使用してユーザオブジェクトのパスワードを変更したとき。
52	ndsLogout	eDirectory からログアウトされたとき。  例：  Novell クライアントからツリーへの接続を切ったとき。
53	ndsAddReplica	サーバパーティションにレプリカが追加されたとき。  例：  ndsconfig を使用してツリーに新規レプリカを追加したとき。
54	ndsRemoveReplica	レプリカが削除されたとき。  例：  ConsoleOne または iManager を使用して、あるサーバからレプリカを削除したとき。
55	ndsSplitPartition	パーティションが分割されたとき。  例：  ConsoleOne または iManager を使用して、パーティションを作成したとき。
56	ndsJoinPartition	ペアレントパーティションにチャイルドパーティションが結合されたとき。  例：  パーティションを作成し、ConsoleOne または iManager を使用して結合したとき。

トラップ 番号	トラップ名	生成される条件
57	ndsChangeReplicaType	パーティションのレプリカタイプが変更されたとき。 例： レプリカタイプをマスタから読み書き用に変更したとき。
58	ndsAddEntry	新規オブジェクトが追加されたとき。 例： ConsoleOne または iManager を使用して、新規オブジェクトを追加したとき。
59	ndsAbortPartitionOp	パーティションに関する処理が中断されたとき。 例： コンテナのパーティション分割処理を実行し、途中で中断したとき。
60	ndsRecvReplicaUpdates	同期処理中に、レプリカ側で更新通知を受け取ったとき。 例： 複数のサーバツリーから成る構成の eDirectory サーバで、レプリカを保持しているサーバに対して更新要求を送った場合。この操作は ConsoleOne や iManager で実行できます。
61	ndsRepairTimeStamps	レプリカのタイムスタンプが修復されたとき。 例： dsrepair (UNIX の場合は ndsrepair、Windows の場合は NSDCons) を使用して、タイムスタンプの DIB 修復処理を実行したとき。
62	ndsSendReplicaUpdates	同期処理中にレプリカが更新されたとき。 例： 複数のサーバツリーから成る構成の eDirectory サーバで、レプリカを保持しているサーバに対して更新要求を送った場合。この操作は ConsoleOne や iManager で実行できます。
63	ndsVerifyPass	パスワードが正しいと確認されたとき。 例： 無効になったパスワードを変更する際、確認のために入力させた旧パスワードが正しいと確認されたとき。
64	ndsBackupEntry	エントリがバックアップされたとき。 例： dsbackup ユーティリティ (UNIX の場合は ndsbackup、Windows の場合は NSDCons) を使用して、eDirectory オブジェクトをバックアップしたとき。

トラップ 番号	トラップ名	生成される条件
65	ndsRestoreEntry	<p>エントリが復元されたとき。</p> <p>例：</p> <p>dsbackup ユーティリティ (UNIX の場合は ndsbackup、Windows の場合は NDSCons) を使用して、バックアップされていた eDirectory オブジェクトを復元したとき。</p>
66	ndsDefineAttributeDef	<p>スキーマに属性定義が追加されたとき。</p> <p>例：</p> <p>eDirectory ツリースキーマに属性定義を追加して拡張したとき。ZENWorks<sup>®</sup>、NMAST<sup>™</sup> などといった eDirectory 用アプリケーションをインストールすると、スキーマは拡張されます。また、ConsoleOne や iManager、UNIX 用スキーマ拡張ユーティリティ ndssch でも拡張できます。</p>
67	ndsRemoveAttributeDef	<p>スキーマから属性定義が削除されたとき。</p> <p>例：</p> <p>eDirectory ツリースキーマから属性定義を削除したとき。属性は、ConsoleOne や iManager、UNIX 用スキーマ拡張ユーティリティ ndssch を使用して削除できます。</p>
68	ndsRemoveClassDef	<p>スキーマからクラス定義が削除されたとき。</p> <p>例：</p> <p>eDirectory ツリースキーマからオブジェクトクラス定義を削除したとき。オブジェクトクラスは、ConsoleOne や iManager、UNIX 用スキーマ拡張ユーティリティ ndssch を使用して削除できます。</p>
69	ndsDefineClassDef	<p>スキーマにクラス定義が追加されたとき。</p> <p>例：</p> <p>eDirectory ツリースキーマにクラス定義を追加して拡張したとき。ZENWorks、NMAST などといった eDirectory 用アプリケーションをインストールすると、スキーマは拡張されます。また、ConsoleOne や iManager、UNIX 用スキーマ拡張ユーティリティ ndssch でも拡張できます。</p>
70	ndsModifyClassDef	<p>クラス定義が変更されたとき。</p> <p>例：</p> <p>既存のオブジェクトクラスや属性定義を変更したとき。</p>
71	ndsResetDSCounters	eDirectory に内蔵されたカウンタがリセットされたとき。
72	ndsRemoveEntryDir	エントリに関連づけられたディレクトリが削除されたとき。

トラップ 番号	トラップ名	生成される条件
73	ndsCompAttributeValue	属性値が比較されたとき。  例：  属性値を他のオブジェクトの属性値と比較したとき。ユーザオブジェクトを対象とするLDAP検索により、入力された値と電話番号が一致するかどうか検査するような場合。
74	ndsOpenStream	ストリーム属性がオープンまたはクローズされたとき。  例：  読み込み用または書き出し用としてストリームを生成した、あるいはオープンしたとき。ユーザオブジェクトのログインスクリプトを作成したとき。DIB ディレクトリ以下にファイルが生成される結果、このトラップが発生します。
75	ndsListSubordinates	コンテナオブジェクトに対して、それに含まれるエントリのリストを取得する処理が実行されたとき。この処理では、コンテナオブジェクトの直下にあるエントリのみが検索の対象となります。  例：  ConsoleOne または iManager を使用して、コンテナオブジェクトをクリックすることにより、これに含まれるオブジェクトを一覧表示しようとしたとき。
76	ndsListContainerClasses	エントリに対して、これを含めることができるクラスのリストを取得する処理が実行されたとき。  例：  あるオブジェクトについて、これを含めることができるクラス(コンテナクラス)を一覧表示しようとしたとき。  ユーザオブジェクトに対してクエリを行った場合、そのオブジェクトを含めることができるコンテナクラスは、組織、部署、およびドメインクラスです。
77	ndsInspectEntry	エントリの検査処理が実行されたとき。  例：  あるエントリについて、これまでにエラーが発生したことがあるかどうかを検査しようとしたとき。バックグラウンドで eDirectory のフラットクリーナ処理を実行する際にこのイベントが発生する結果、トラップが送られることとなります。
78	ndsResendEntry	エントリの再送信処理が実行されたとき。  例：  レプリカの作成処理中に、エントリの再送信が起こったとき。サーバ間の接続に問題があり、オブジェクトの送信に失敗したような場合に起こります。

トラップ 番号	トラップ名	生成される条件
79	ndsMutateEntry	<p>エントリの変換処理が実行されたとき。</p> <p>例：</p> <p>バインダリオブジェクトクラスからユーザオブジェクトクラスに変換したとき。</p>
80	ndsMergeEntries	<p>2つのエントリがマージされたとき。</p> <p>例：</p> <p>2つのユーザオブジェクトをマージしたとき。 Entry2(ndsEntryName2) を Entry(ndsEntryName) にマージしたような場合。</p>
81	ndsMergeTree	<p>2つの eDirectory ツリーがマージされたとき。</p> <p>例：</p> <p>dsmerge(UNIX の場合は ndsmerge、Windows の場合は NDSCons) を使用して、2つの eDirectory ツリーをマージしたとき。</p>
82	ndsCreateSubref	<p>サブオーディネートリファレンスが生成されたとき。</p> <p>例：</p> <p>チャイルドパーティションのレプリカをサーバから削除したとき。自動的にサブオーディネートリファレンスのレプリカが生成され、その結果トラップが発生します。</p>
83	ndsListPartitions	<p>パーティションのリスト取得処理が実行されたとき。</p> <p>例：</p> <p>ConsoleOne や iManager を使い、パーティションビューやスキーマビューで eDirectory サーバオブジェクトをクリックすることにより、サーバ上のパーティションを一覧表示したとき。</p>
84	ndsReadAttribute	<p>属性値が読み出されたとき。</p> <p>例：</p> <p>ツリーの検索処理を実行したとき。</p>
85	ndsReadReferences	<p>エントリの参照が読み出されたとき。</p>
86	ndsUpdateReplica	<p>パーティションレプリカに対して、レプリカの更新処理が実行されたとき。</p> <p>例：</p> <p>あるサーバからユーザを削除したとき。他のサーバ上のレプリカでもこれを削除しようとするため、更新処理が起こります。</p>

トラップ 番号	トラップ名	生成される条件
87	ndsStartUpdateReplica	パーティションレプリカに対して、レプリカの更新開始処理が開始されたとき。  例：  あるサーバからユーザを削除したとき。他のサーバ上のレプリカでもこれを削除しようとするため、更新処理が起こります。
88	ndsEndUpdateReplica	パーティションレプリカに対して、レプリカの更新処理が実行され、終了したとき。  例：  あるサーバからユーザを削除したとき。他のサーバ上のレプリカでもこれを削除しようとするため、更新処理が起こります。
89	ndsSyncPartition	パーティションレプリカに対して、パーティションの同期処理が実行されたとき。  例：  あるパーティションからユーザを削除したとき。ndstrace を使うと同期処理の状況を見ることができます。
90	ndsSyncSchema	ルートのマスタレプリカが、そのスキーマをサーバと同期させる要求を受け取ったとき。  例：  ConsoleOne から [ウィザード] > [スキーマ] と操作するか、LDAP ツールまたは ndssch ユーティリティを使用して新規クラスを追加したとき。
91	ndsCreateBackLink	バックリンクが生成されたとき。ローカルには存在しないオブジェクトが参照されると、バックリンクが生成されます。 例：複数のサーバから成る構成で、いくつかのユーザを含むパーティションを生成したとします。サーバの1つからこのパーティションを削除すると、サブオーディネートリファレンスが生成されます。このとき、削除されたパーティションに存在していたユーザを参照するバックリンクが生成されません。
92	ndsCheckConsoleOperator	バックリンクにより、コンソールオペレータの権利が検査されたとき。  このトラップは NetWare でのみ発生します。
93	ndsChangeTreeName	ツリー名が変更されたとき。  例：  マージユーティリティ dsmerge/ndsmerge を使用してツリー名を変更したとき。
94	ndsStartJoinPartition	パーティションの結合処理が始まったとき。  例：  ConsoleOne または LDAP ツールを使用して、パーティションを結合したとき。

トラップ 番号	トラップ名	生成される条件
95	ndsAbortJoinPartition	パーティションの結合処理が中断されたとき。  例：  ConsoleOne または LDAP ツールを使用して、パーティションを結合したとき。
96	ndsUpdateSchema	スキーマ更新処理が実行されたとき。  例：  ConsoleOne から [ウィザード] > [スキーマ] と操作するか、LDAP ツールまたは ndssch ユーティリティを使用して新規クラスを追加したとき。
97	ndsStartUpdateSchema	スキーマ更新処理が開始されたとき。  例：  ConsoleOne から [ウィザード] > [スキーマ] と操作するか、LDAP ツールまたは ndssch ユーティリティを使用して新規クラスを追加したとき。
98	ndsEndUpdateSchema	スキーマ更新処理が終了したとき。  例：  ConsoleOne から [ウィザード] > [スキーマ] と操作するか、LDAP ツールまたは ndssch ユーティリティを使用して新規クラスを追加したとき。
99	ndsMoveTree	ツリー移動処理が実行されたとき。  例：  パーティションをあるコンテナから別のコンテナに移動したとき。
100	ndsReloadDS	DS が再ロードされたとき。  このトラップは NetWare でのみ発生します。  例：  dstrace=* と設定したとき。
101	ndsConnectToAddress	特定のアドレスとの間で接続が確立されたとき。  例：  ConsoleOne または iManager を使用してツリーをブラウズしたとき。
102	ndsSearch	検索処理が実行されたとき。  例：  LDAP ツールを使用してツリーに対する ldapsearch を実行したとき。

トラップ 番号	トラップ名	生成される条件
103	ndsPartitionStateChange	パーティションが作成または削除されたとき。  例：  新規パーティションを作成したとき。
104	ndsRemoveBacklink	使われていない外部参照が削除され、該当するオブジェクトを保持しているサーバに対して、バックリンク削除要求が送られたとき。
105	ndsLowLevelJoinPartition	パーティションの結合処理中に、低レベルの結合処理が実行されたとき。  例：  ConsoleOne、iManager、または LDAP ツールを使用して、パーティションを結合したとき。
106	ndsCreateNameBase	eDirectory ネームベースが作成されたとき。
107	ndsChangeSecurityEquals	同等セキュリティ属性が変更されたとき。  例：  ConsoleOne または iManager を使用して、あるユーザの同等セキュリティを変更し、管理者と同等にしたとき。
108	ndsRemoveEntry	eDirectory からエントリが削除されたとき。  例：  ConsoleOne または iManager を使用して、ユーザを削除したとき。
109	ndsCRCFailure	断片化した NCP 要求を構成し直す際に、CRC(冗長巡回検査)エラーが発生したとき。
110	ndsModifyEntry	eDirectory エントリが変更されたとき。  例：  ConsoleOne または iManager を使用して、ユーザの属性を変更したとき。
111	ndsNewSchemaEpoch	スキーマが DSRepair でリセットされたとき。  例：  UNIX 上で、ndsrepair -S-Ad を使用して新しいスキーマエポックを作成したとき。
112	ndsLowLevelSplitPartition	パーティションを作成する際に、低レベル分割処理が実行されたとき。  例：  ConsoleOne、iManager、または LDAP ツールを使用して、パーティションを作成したとき。
113	ndsReplicaInTransition	レプリカが追加または削除されたとき。



トラップ 番号	トラップ名	生成される条件
114	ndsAclModify	<p>オブジェクトのトラスティが変更された、すなわち ACL( アクセス制御リスト )オブジェクトが変更されたとき。</p> <p>例：</p> <p>LDAP ツール、ICE、ConsoleOne、または iManager を使用して、オブジェクトのトラスティを追加、変更、削除したとき。</p>
115	ndsLoginEnable	<p>ユーザアカウントを有効にする要求をサーバから受け取ったとき。</p> <p>例：</p> <p>LDAP ツール、ICE、ConsoleOne、または iManager を使用して、アカウント属性を無効から有効に変更したとき。</p>
116	ndsLoginDisable	<p>ユーザアカウントを無効にする要求をサーバから受け取ったとき。</p> <p>例：</p> <p>LDAP ツール、ICE、ConsoleOne、または iManager を使用して、アカウント属性を有効から無効に変更したとき。</p>
117	ndsDetectIntruder	<p>不正侵入を検出したため、ユーザアカウントがロックされたとき。</p> <p>例：</p> <p>LDAP ツール、ICE、ConsoleOne、または iManager を使用して Intruder 属性を設定することにより、ユーザアカウントをロックしたとき。</p>
2001	ndsServerStart	<p>サブエージェントが eDirectory サーバに、正常に再接続できたとき。このトラップには 2 つの変数が含まれています。</p> <ul style="list-style-type: none"> <li>◆ ndsTrapTime: サブエージェントが eDirectory サーバに再接続した時刻を、1970 年 1 月 1 日午前 0 時 ( 万国標準時 ) からの経過秒数で表します。</li> <li>◆ ndsServerName: サブエージェントが再接続した eDirectory サーバを表します。</li> </ul> <p>例：</p> <p>サブエージェントが稼動したままの状態であった eDirectory サーバを停止し、再び起動したとき。</p>

トラップ 番号	トラップ名	生成される条件
2002	ndsServerStop	<p>サブエージェントと eDirectory サーバとの接続が失われたとき。このトラップには 2 つの変数が含まれています。</p> <ul style="list-style-type: none"> <li>ndsTrapTime:サブエージェントがeDirectoryサーバと切断された時刻を、1970 年 1 月 1 日午前 0 時 (万国標準時) からの経過秒数で表します。</li> <li>ndsServerName: サブエージェントがそれまで接続されていた eDirectory サーバを表します。</li> </ul> <p>例:</p> <p>サブエージェントが稼動中に eDirectory サーバを停止したとき。</p>

## トラップに関する設定

トラップに関する設定の手順はプラットフォームによって異なります。

プラットフォーム	ユーティリティ
NetWare	dssnmppsa
Windows	ndssnmpcfg
UNIX	ndssnmpconfig

### NetWare の場合

NetWare では、`dssnmppsa` を使用してトラップに関する設定を行います。これは `sys:\etc\` ディレクトリにあります。このユーティリティの機能としては、トラップの有効 / 無効の切り替え、各トラップ間の時間間隔の設定、デフォルトの時間間隔の設定、操作に失敗した場合のトラップの有効化、すべてのトラップのリスト表示などがあります。

`dssnmppsa` の使用方法に関するヘルプを表示するには、コマンドラインで「`help dssnmppsa`」と入力します。

使用法:

`dssnmppsa trap commands`

NetWare のトラップコマンドについては、[367 ページの「NetWare のトラップコマンド」](#)を参照してください。

## NetWare のトラップコマンド

トラップコマンド	説明	使用法
DISABLE	トラップを無効にするコマンド。NMS は、トラップが送られてきても受け取らないようになります。	<pre>dssnmppsa "DISABLE trapSpec"</pre> <p><i>trapSpec</i> は次のいずれかの形式です。</p> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を無効にする場合 :</p> <pre>dssnmppsa "DISABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて無効にする場合 :</p> <pre>dssnmppsa "DISABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を無効にする場合 :</p> <pre>dssnmppsa "DISABLE 20-29"</pre> <p>トラップをすべて無効にする場合 :</p> <pre>dssnmppsa "DISABLE ALL"</pre>
ENABLE	トラップを有効にするコマンド。NMS は、送られてきたトラップを受け取るようになります。	<pre>dssnmppsa "ENABLE trapSpec"</pre> <p><i>trapSpec</i> は次のいずれかの形式で指定してください。</p> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を有効にする場合 :</p> <pre>dssnmppsa "ENABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて有効にする場合 :</p> <pre>dssnmppsa "ENABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を有効にする場合 :</p> <pre>dssnmppsa "ENABLE 20-29"</pre> <p>トラップをすべて有効にする場合 :</p> <pre>dssnmppsa "ENABLE ALL"</pre>

トラップコマンド 説明	使用法
<p>INTERVAL 時間間隔を設定する、または表示するためのコマンド。</p> <p>ここでいう時間間隔とは、同じトラップを繰り返し送る場合に、何秒間の間隔をおくかを表すものです。</p> <p>0 ~ 2,592,000 の範囲 (秒単位) で設定します。</p> <p>この範囲外の値を指定した場合、デフォルトの時間間隔が指定されたものとみなします。</p> <p>設定値を 0 とすれば、トラップがすべて送信されます。</p>	<p>時間間隔の設定値を表示する場合： dssnmpsa "213,240,79 INTERVAL"</p> <p>複数のトラップについて時間間隔を設定する場合 ( 次の例では 12 番、17 番、101 番トラップについて 5 秒と設定 )： dssnmpsa "12 17 101 INTERVAL 5"</p> <p>デフォルトの時間間隔を表示する場合： dssnmpsa "DEFAULT INTERVAL"</p> <p>デフォルトの時間間隔を設定する場合： dssnmpsa "DEFAULT INTERVAL = 10"</p>
<p>LIST ある条件を満たすトラップ番号を一覧表示するコマンド。</p>	<p>dssnmpsa LIST <i>trapSpec</i></p> <p><i>trapSpec</i> には、トラップ番号のほか、以下に述べるキーワードで条件を指定できます。</p> <p>ALL、ENABLED、DISABLED、FAILED、またはこれらを論理演算子でつないだもの。</p> <p>例：</p> <p>有効なトラップをすべて、名称を添えて表示： dssnmpsa LIST ENABLED</p> <p>無効なトラップをすべて、名称を添えて表示： dssnmpsa LIST DISABLED</p> <p>119 種類すべてのトラップについて、名称を添えて表示： dssnmpsa LIST ALL</p> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) のトラップについて、名称を添えて表示： dssnmpsa LIST ID = 12,224,300</p> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) 以外のトラップをすべて、名称を添えて表示： dssnmpsa LIST ID != 12,224,300</p> <p>有効なエラートラップをすべて、名称を添えて表示： dssnmpsa LIST FAILED</p>

トラップコマンド	説明	使用法
READ_CFG	<p>設定ファイルndstrap.cfgを参照して、ディレクトリ構成を再設定するコマンド。</p> <p>環境設定ファイルの設定に加えられた変更はすべて有効になります。いくつものコマンドを ndstrap.cfg に記述しておき、ひとまとめにして実行する、という使い方を主として想定したユーティリティです。</p> <p>ndstrap.cfg は sys:\etc\ 以下にあります。</p> <p>ndstrap.cfg ファイルでは、トラップの環境設定に使用されるオペレーショナルパラメータを設定し、SNMP トラップの操作を設定する方法を指定します。このファイルは、READ_CFG コマンドにより起動される、トラップの環境設定ユーティリティ ndssnmconfig が参照します。</p>	<pre>dssnmppsa "READ_CFG"</pre>
FAILURE	<p>エラートラップをすべて表示するコマンド。</p> <p>エラートラップとは、イベントの失敗時に生成されるトラップのことです。</p> <p><b>注:</b> エラートラップをいったん無効にし、「enable trapid」コマンドで再び有効にすると、エラートラップではなく、正常に処理されたことを表すトラップとして扱われるようになります。</p>	<pre>dssnmppsa "FAILURE trapSpec"</pre> <p>trapSpec には、トラップ番号をコンマまたはスペースで区切って指定するほか、キーワード「ALL」や論理式を指定できます。</p> <p>例：</p> <p>複数のトラップをエラートラップと設定：</p> <pre>dssnmppsa "FAILURE 10,11,100"</pre> <p>指定した番号以外のすべてのトラップをエラートラップと設定：</p> <pre>dssnmppsa "FAILURE ID != 24,30"</pre> <p>すべてのトラップをエラートラップと設定：</p> <pre>dssnmppsa "FAILURE ALL"</pre>

## Windows の場合

Windows では、ndssnmppcfg を使用してトラップに関する設定を行います。このユーティリティは *install\_path\snmp\* ディレクトリにあります。このユーティリティの機能としては、トラップの有効 / 無効の切り替え、各トラップ間の時間間隔の設定、デフォルトの時間間隔の設定、操作に失敗した場合のトラップの有効化、すべてのトラップのリスト表示などがあります。

使用法：

```
ndssnmppcfg -h [hostname[:port]] -p password -a userFDN -c command
```

パラメータ	説明
-h	DNS ホスト名または IP アドレス
-p	認証に使う userFDN パスワード
-a	管理者権を持つユーザの完全識別名
-c	トラップコマンド (370 ページの「Windows のトラップコマンド」を参照)

## Windows のトラップコマンド

トラップコマンド	説明	使用法
DISABLE	トラップを無効にするコマンド。NMS は、トラップが送られてきても受け取らないようになります。	<p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を無効にする場合 :</p> <pre>ndssnmpcfg "DISABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて無効にする場合 :</p> <pre>ndssnmpcfg "DISABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を無効にする場合 :</p> <pre>ndssnmpcfg "DISABLE 20-29"</pre> <p>トラップをすべて無効にする場合 :</p> <pre>ndssnmpcfg "DISABLE ALL"</pre>
ENABLE	トラップを有効にするコマンド。NMS は、送られてきたトラップを受け取るようになります。	<p><code>ndssnmpcfg "ENABLE <i>trapSpec</i>"</code></p> <p><i>trapSpec</i> は次のいずれかの形式で指定してください。</p> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を有効にする場合 :</p> <pre>ndssnmpcfg "ENABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて有効にする場合 :</p> <pre>ndssnmpcfg "ENABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を有効にする場合 :</p> <pre>ndssnmpcfg "ENABLE 20-29"</pre> <p>トラップをすべて有効にする場合 :</p> <pre>ndssnmpcfg "ENABLE ALL"</pre>

トラップコマンド	説明	使用法
INTERVAL	<p>時間間隔を設定する、または表示するためのコマンド。</p> <p>ここでいう時間間隔とは、同じトラップを繰り返し送る場合に、何秒間の間隔をおくかを表すものです。</p> <p>0 ~ 2,592,000 の範囲 ( 秒単位 ) で設定します。</p> <p>この範囲外の値を指定した場合、デフォルトの時間間隔が指定されたものとみなします。</p> <p>設定値を 0 とすれば、トラップがすべて送信されます。</p>	<p>時間間隔の設定値を表示する場合：</p> <pre>ndssnmpcfg "213,240,79 INTERVAL"</pre> <p>複数のトラップについて時間間隔を設定する場合 ( 次の例では 12 番、17 番、101 番トラップについて 5 秒と設定 ):</p> <pre>ndssnmpcfg "12 17 101 INTERVAL 5"</pre> <p>デフォルトの時間間隔を表示する場合：</p> <pre>ndssnmpcfg "DEFAULT INTERVAL"</pre> <p>デフォルトの時間間隔を設定する場合：</p> <pre>ndssnmpcfg "DEFAULT INTERVAL=10"</pre>
LIST	<p>ある条件を満たすトラップ番号を一覧表示するコマンド。</p>	<pre>ndssnmpcfg LIST trapSpec</pre> <p><i>trapSpec</i> には、トラップ番号のほか、以下に述べるキーワードで条件を指定できます。</p> <p>ALL、ENABLED、DISABLED、FAILED、またはこれらを論理演算子でつないだもの。</p> <p>例：</p> <p>有効なトラップをすべて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST ENABLED</pre> <p>無効なトラップをすべて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST DISABLED</pre> <p>119 種類すべてのトラップについて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST ALL</pre> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) のトラップについて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST ID = 12,224,300</pre> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) 以外のトラップをすべて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST ID != 12,224,300</pre> <p>有効なエラートラップをすべて、名称を添えて表示：</p> <pre>ndssnmpcfg LIST FAILED</pre>

トラップコマンド 説明	使用法
<p>READ_CFG 設定ファイルndstrap.cfgを参照して、ディレクトリ構成を再設定するコマンド。</p> <p>環境設定ファイルの設定に加えられた変更はすべて有効になります。いくつものコマンドをndstrap.cfgに記述しておき、ひとまとめにして実行する、という使い方を主として想定したユーティリティです。</p> <p>ファイルndstrap.cfgはディレクトリinstall directory\SNMPにあります。</p> <p>ndstrap.cfgファイルでは、トラップの環境設定に使用されるオペレーショナルパラメータを設定し、SNMPトラップの操作を設定する方法を指定します。このファイルは、READ_CFGコマンドにより起動される、トラップの環境設定ユーティリティndssnmpcfgが参照します。</p>	<p>ndssnmpcfg "READ_CFG"</p>
<p>FAILURE エラートラップをすべて表示するコマンド。</p> <p>エラートラップとは、イベントの失敗時に生成されるトラップのことです。</p> <p>注：エラートラップをいったん無効にし、「enable trapid」コマンドで再び有効にすると、エラートラップではなく、正常に処理されたことを表すトラップとして扱われるようになります。</p>	<p>ndssnmpcfg "FAILURE trapSpec"</p> <p>trapSpecには、トラップ番号をコンマまたはスペースで区切って指定するほか、キーワード「ALL」や論理式を指定できます。</p> <p>例：</p> <p>複数のトラップをエラートラップと設定：</p> <p>ndssnmpcfg "FAILURE 10,11,100"</p> <p>指定した番号以外のすべてのトラップをエラートラップと設定：</p> <p>ndssnmpcfg "FAILURE ID != 24,30"</p> <p>すべてのトラップをエラートラップと設定：</p> <p>ndssnmpcfg "FAILURE ALL"</p>

## UNIX

UNIX では、ndssnmconfig を使用してトラップに関する設定を行います。これは /etc/ndssnmp/ ディレクトリにあります。このユーティリティの機能としては、トラップの有効/無効の切り替え、各トラップ間の時間間隔の設定、デフォルトの時間間隔の設定、操作に失敗した場合のトラップの有効化、すべてのトラップのリスト表示などがあります。

使用法：

```
ndssnmconfig -h [hostname[:port]] -p password -a userFDN -c command
```



パラメータ	説明
-h	DNS ホスト名または IP アドレス
-p	認証に使う userFDN パスワード
-a	管理者権を持つユーザの完全識別名
-c	トラップコマンド ( <a href="#">373 ページの「UNIX のトラップコマンド」</a> を参照)

## UNIX のトラップコマンド

トラップコマンド	説明	使用法
DISABLE	トラップを無効にするコマンド。NMS は、トラップが送られてきても受け取らないようになります。	<p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を無効にする場合 :</p> <pre>ndssnmpconfig "DISABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて無効にする場合 :</p> <pre>ndssnmpconfig "DISABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を無効にする場合 :</p> <pre>ndssnmpconfig "DISABLE 20-29"</pre> <p>トラップをすべて無効にする場合 :</p> <pre>ndssnmpconfig "DISABLE ALL"</pre>
ENABLE	トラップを有効にするコマンド。NMS は、送られてきたトラップを受け取るようになります。	<pre>ndssnmpconfig "ENABLE trapSpec"</pre> <p><i>trapSpec</i> は次のいずれかの形式で指定してください。</p> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) を有効にする場合 :</p> <pre>ndssnmpconfig "ENABLE 10, 11, 100"</pre> <p>特定のトラップ ( 次の例では 10 番、11 番、100 番 ) 以外をすべて有効にする場合 :</p> <pre>ndssnmpconfig "ENABLE ID != 10, 11, 100"</pre> <p>ある範囲の番号のトラップ ( 次の例では 20 ~ 29 番 ) を有効にする場合 :</p> <pre>ndssnmpconfig "ENABLE 20-29"</pre> <p>トラップをすべて有効にする場合 :</p> <pre>ndssnmpconfig "ENABLE ALL"</pre>

トラップコマンド	説明	使用法
INTERVAL	<p>時間間隔を設定する、または表示するためのコマンド。</p> <p>ここでいう時間間隔とは、同じトラップを繰り返し送る場合に、何秒間の間隔をおくかを表すものです。</p> <p>0 ~ 2,592,000 の範囲 ( 秒単位 ) で設定します。</p> <p>この範囲外の値を指定した場合、デフォルトの時間間隔が指定されたものとみなします。</p> <p>設定値を 0 とすれば、トラップがすべて送信されます。</p>	<p>時間間隔の設定値を表示する場合 :</p> <pre>ndssnmpconfig "213,240,79 INTERVAL"</pre> <p>複数のトラップについて時間間隔を設定する場合 ( 次の例では 12 番、17 番、101 番トラップについて 5 秒と設定 ):</p> <pre>ndssnmpconfig "12 17 101 INTERVAL 5"</pre> <p>デフォルトの時間間隔を表示する場合 :</p> <pre>ndssnmpconfig "DEFAULT INTERVAL"</pre> <p>デフォルトの時間間隔を設定する場合 :</p> <pre>ndssnmpconfig "DEFAULT INTERVAL=10"</pre>
LIST	<p>ある条件を満たすトラップ番号を一覧表示するコマンド。</p>	<pre>ndssnmpconfig LIST &lt;trapSpec&gt;</pre> <p><i>trapSpec</i> には、トラップ番号のほか、以下に述べるキーワードで条件を指定できます。</p> <p>ALL、ENABLED、DISABLED、FAILED、またはこれらを論理演算子でつないだもの。</p> <p>例 :</p> <p>有効なトラップをすべて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST ENABLED</pre> <p>無効なトラップをすべて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST DISABLED</pre> <p>119 種類すべてのトラップについて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST ALL</pre> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) のトラップについて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST ID = 12,224,300</pre> <p>特定の番号 ( 次の例では 12 番、224 番、300 番 ) 以外のトラップをすべて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST ID != 12,224,300</pre> <p>有効なエラートラップをすべて、名称を添えて表示 :</p> <pre>ndssnmpconfig LIST FAILED</pre>

トラップコマンド	説明	使用法
READ_CFG	<p>設定ファイル <code>ndstrap.cfg</code> を参照して、ディレクトリ構成を再設定するコマンド。</p> <p>環境設定ファイルの設定に加えられた変更はすべて有効になります。いくつものコマンドを <code>ndstrap.cfg</code> に記述しておき、ひとまとめにして実行する、という使い方を主として想定したユーティリティです。</p> <p>ファイル <code>ndstrap.cfg</code> はディレクトリ <code>/etc/ndssnmp/</code> にあります。</p> <p><code>ndstrap.cfg</code> ファイルでは、トラップの環境設定に使用されるオペレーショナルパラメータを設定し、SNMP トラップの操作を設定する方法を指定します。このファイルは、<code>READ_CFG</code> コマンドにより起動される、トラップの環境設定ユーティリティ <code>ndssnmpcfg</code> が参照します。</p>	<p><code>ndssnmpconfig "READ_CFG"</code></p>
FAILURE	<p>エラートラップをすべて表示するコマンド。</p> <p>エラートラップとは、イベントの失敗時に生成されるトラップのことです。</p> <p>注：エラートラップをいったん無効にし、「<code>enable trapid</code>」コマンドで再び有効にすると、エラートラップではなく、正常に処理されたことを表すトラップとして扱われるようになります。</p>	<p><code>ndssnmpconfig "FAILURE trapSpec"</code></p> <p><code>trapSpec</code> には、トラップ番号をコンマまたはスペースで区切って指定するほか、キーワード「ALL」や論理式を指定できます。</p> <p>例：</p> <p>複数のトラップをエラートラップと設定：</p> <p><code>ndssnmpconfig "FAILURE 10,11,100"</code></p> <p>指定した番号以外のすべてのトラップをエラートラップと設定：</p> <p><code>ndssnmpconfig "FAILURE ID != 24,30"</code></p> <p>すべてのトラップをエラートラップと設定：</p> <p><code>ndssnmpconfig "FAILURE ALL"</code></p>

## 統計情報

- ◆ [376 ページの「ndsDbCache」](#)
- ◆ [376 ページの「ndsDbConfig」](#)
- ◆ [377 ページの「ndsProtoIfOps」](#)
- ◆ [379 ページの「ndsServerInt」](#)

## ndsDbCache

ディレクトリ以下にある管理対象オブジェクト	説明
ndsDbSrvApplIndex	eDirectory サーバアプリケーションを一意に識別するインデックス。
ndsDbDibSize	eDirectory データベースの容量 (KB 単位)。
ndsDbBlockSize	eDirectory データベースのブロック容量 (KB 単位)。
ndsDbEntryCacheMaxSize	エントリキャッシュの最大容量 (KB 単位)。
ndsDbBlockCacheMaxSize	ブロックキャッシュの最大容量 (KB 単位)。
ndsDbEntryCacheCurrentSize	現在のエントリキャッシュ容量。
ndsDbBlockCacheCurrentSize	現在のブロックキャッシュ容量。
ndsDbEntryCacheCount	キャッシュ内のエントリ数。
ndsDbBlockCacheCount	キャッシュ内のブロック数。
ndsDbEntryCacheOldVerCount	キャッシュ内に残っている旧バージョンのエントリ数。
ndsDbBlockCacheOldVerCount	キャッシュ内に残っている旧バージョンのブロック数。
ndsDbEntryCacheOldVerSize	旧バージョンのエントリキャッシュ容量。
ndsDbBlockCacheOldVerSize	旧バージョンのブロックキャッシュ容量。
ndsDbEntryCacheHits	キャッシュ内のエントリがヒットした回数。
ndsDbBlockCacheHits	キャッシュ内のブロックがヒットした回数。
ndsDbEntryCacheHitLooks	キャッシュ内のエントリがヒットするかどうか試みられた回数。
ndsDbBlockCacheHitLooks	キャッシュ内のブロックがヒットするかどうか試みられた回数。
ndsDbEntryCacheFaults	キャッシュ内のエントリがヒットしなかった回数。
ndsDbBlockCacheFaults	キャッシュ内のブロックがヒットしなかった回数。
ndsDbEntryCacheFaultLooks	キャッシュ内のエントリがヒットしないかどうか試みられた回数。
ndsDbBlockCacheFaultLooks	キャッシュ内のブロックがヒットしないかどうか試みられた回数。

## ndsDbConfig

ディレクトリ以下にある管理対象オブジェクト	説明
ndsDbCfgSrvApplIndex	eDirectory サーバアプリケーションを一意に識別するインデックス。

ディレクトリ以下にある管理対象オブジェクト	説明
ndsDbCfgDynamicCacheAdjust	動的キャッシュ調整が有効かどうか。 0 = 無効 1 = 有効
ndsDbCfgDynamicCacheAdjustPercent	動的キャッシュ調整に、空きメモリの何 % を割り当てるか。
ndsDbCfgDynamicCacheAdjustMin	動的キャッシュ調整に使う最小容量。キャッシュ容量制限を KB 単位で表したものの。
ndsDbCfgDynamicCacheAdjustMinToLeave	動的キャッシュ調整に使う最小容量のうち、利用可能なメモリから差し引く容量 (KB 単位)。
ndsDbCfgHardLimitCacheAdjust	キャッシュ調整に割り当てるメモリ容量にハード制限を設定するかどうか。 0 = 無効 1 = 有効
ndsDbCfgHardLimitCacheAdjustMax	キャッシュの最大容量 (KB 単位)。これはハードメモリ制限を表します。
ndsDbCfgBlockCachePercent	ブロックキャッシュに割り当てる比率。
ndsDbCfgCacheAdjustInterval	キャッシュ調整を行う時間間隔 (秒単位)。
ndsDbCfgCacheCleanupInterval	キャッシュのクリーンアップを行う時間間隔(秒単位)。
ndsDbCfgPermanentSettings	常時接続の設定が有効かどうか。 0 = 無効 1 = 有効

## ndsProtolfOps

ディレクトリ以下にある管理対象オブジェクト	説明
ndsProtolfSrvApplIndex	eDirectory サーバアプリケーションを一意に識別するインデックス。
ndsProtolfIndex	eDirectory サーバのプロトコルインタフェースに対応するエントリを一意に識別するインデックス。
ndsProtolfDescription	DS プロトコルインタフェースに使うポート番号。
ndsProtolfUnauthBinds	認証を省略した匿名バインド要求を受け取った回数。
ndsProtolfSimpleAuthBinds	バインド要求のうち、簡易認証手続きにより認証に成功したものの回数。簡易認証手続きとは、パスワードを暗号化して、またはクリアテキストのまま送ることにより行うものです。

ディレクトリ以下にある管理対象オブジェクト	説明
ndsProtolfStrongAuthBinds	バインド要求のうち、強度の高い認証手続きである SASL および X.500 の認証に成功したものの回数。外部認証手続きによるものも数に含まれます。
ndsProtolfBindSecurityErrors	バインド要求のうち、認証手続きが適切でない、あるいは認証情報が無効であるために拒否したものの回数。
ndsProtolfInOps	DUA( ディレクトリユーザエージェント ) または他の eDirectory サーバから受け取った要求の回数。
ndsProtolfReadOps	受け取った読み出し要求の数。
ndsProtolfCompareOps	受け取った比較要求の数。
ndsProtolfAddEntryOps	受け取ったエントリ追加要求の数。
ndsProtolfRemoveEntryOps	受け取ったエントリ削除要求の数。
ndsProtolfModifyEntryOps	受け取ったエントリ変更要求の数。
ndsProtolfModifyRDNops	受け取った RDN( 相対識別名 ) 変更要求の数。
ndsProtolfListOps	受け取ったリスト要求の数。
ndsProtolfSearchOps	受け取った検索要求 ( ベースオブジェクト検索、1 レベル検索、サブツリー全体の検索 ) の数。
ndsProtolfOneLevelSearchOps	受け取った 1 レベル検索要求の数。
ndsProtolfWholeSubtreeSearchOps	受け取ったサブツリー全体の検索要求の数。
ndsProtolfExtendedOps	拡張処理の回数。
ndsProtolfReferrals	処理要求に応じて返した参照の個数。
ndsProtolfChainings	この eDirectory サーバから他の eDirectory サーバに転送した処理の数。
ndsProtolfSecurityErrors	受け取った要求のうち、セキュリティ保護方針に合致しなかったものの数。
ndsProtolfErrors	受け取った要求のうち、エラーのため応じなかったものの数。ただしセキュリティ保護エラー、参照エラーを除きます。一部でも処理ができたものは数に含めません。たとえば、名前づけや更新、属性、サービスに関連するエラーがあります。
ndsProtolfReplicationUpdatesIn	eDirectory サーバから取得した、または受け取ったレプリカ作成更新の数。
ndsProtolfReplicationUpdatesOut	eDirectory サーバに送った、または検知されたレプリカ作成更新の数。
ndsProtolfInBytes	インタフェース上の受信トラフィック ( バイト単位 )。DUA( ディレクトリユーザエージェント ) からの要求、他の eDirectory サーバからの応答などといったトラフィックがこれに当たります。

ディレクトリ以下にある管理対象オブジェクト	説明
ndsProtolfOutBytes	インタフェース上の送信トラフィック ( バイト単位 )。DUA や eDirectory サーバへの応答、他の eDirectory サーバへの要求などといったトラフィックがこれに当たります。

## ndsServerInt

ディレクトリ以下にある管理対象オブジェクト	説明
ndsSrvIntSrvApplIndex	eDirectory サーバアプリケーションを一意に識別するインデックス。
ndsSrvIntProtolfIndex	eDirectory サーバのプロトコルインタフェースに対応するエントリを一意に識別するインデックス。
ndsSrvIntIndex	このオブジェクトは ndsSrvIntSrvApplIndex および ndsSrvIntProtolfIndex と組み合わせて使います。ndsSrvIntSrvApplIndex で示される eDirectory サーバと、特別なプロトコルを使うピア eDirectory サーバとの間でやり取りされる情報を格納した、仮想的な行を一意的に識別するためのキーとなります。
ndsSrvIntURL	ピア eDirectory サーバの URL。
ndsSrvIntTimeOfCreation	この行が作成された時刻。1970 年 1 月 1 日午前 0 時 ( 万国標準時 ) からの経過秒数で表します。
ndsSrvIntTimeOfLastAttempt	ピア eDirectory サーバとの接続を試みた直近の時刻。1970 年 1 月 1 日午前 0 時 ( 万国標準時 ) からの経過秒数で表します。
ndsSrvIntTimeOfLastSuccess	ピア eDirectory サーバと正常に接続できた直近の時刻。1970 年 1 月 1 日午前 0 時 ( 万国標準時 ) からの経過秒数で表します。
ndsSrvIntFailuresSinceLastSuccess	ピア eDirectory サーバと正常に接続した直近の時刻以降に発生したエラーの数。まだ一度も正常に接続できていない場合は、このエントリが作成されて以来のエラー数。
ndsSrvIntFailures	このエントリが作成されて以来、ピア eDirectory サーバとの接続に失敗した回数。
ndsSrvIntSuccesses	このエントリが作成されて以来、ピア eDirectory サーバとの接続に成功した回数。

## トラブルシューティング

問題を迅速に解決できるよう、実際に発生したエラー状況、その解決に役立つ情報が、ログファイルに記録されています。

詳しくは [545 ページ](#) の「[SNMP のトラブルシューティング](#)」を参照してください。

プラットフォーム	サブエージェント	サーバ	マスタ
Windows NT / 2000	<i>install_directory</i> \nds\ds snmpsa.log	<i>install_directory</i> \nds\ dssnmprsv.log	該当なし
Solaris	/var/nds/ndssnmpsa.log	/var/nds/ndsd.log	/var/adm/messages
Linux	/var/nds/ndssnmpsa.log	/var/nds/ndsd.log	/var/log/messages
AIX	/var/nds/ndssnmpsa.log	/var/nds/ndsd.log	/var/adm/messages
HP-UX	/var/nds/ndssnmpsa.log	/var/nds/ndsd.log	net-snmp-5.0.8 マスタ エージェント :/usr/adm/ snmpd.log  NAA エージェント :/ var/adm/snmpd.log



# 14 Novell eDirectory のバックアップと復元

Novell® eDirectory™ には、レプリカ作成による障害対策機能が組み込まれています。eDirectory ツリーに属するサーバに障害があっても、別のサーバがサービスを続行できます。この意味でレプリカ作成機能は、最も重要な保護機能と位置づけられています。

ただし 1 台だけのサーバで運用している環境では、レプリカ作成は不可能です。また、レプリカを作成していても、完全に復元するのは困難な場合もあります。ハードウェア障害など、機器が破損したときや、火事や水害で何台ものサーバが動かなくなってしまう場合などです。各サーバの eDirectory をバックアップしておけば、ネットワークの耐障害性を高めることとなります。

eDirectory 8.7 には、Backup eMTool という新しいバックアップ / 復元ユーティリティが付属するようになりました。これを使用すると、サーバ単位で eDirectory データベースをバックアップすることができます。これには次の利点があります。

- ◆ **どのプラットフォームでも同じツールで操作可能。**
- ◆ **稼動中でもバックアップ可能。**eDirectory データベースを停止することなく、そのまま完全なバックアップを取ることができます。
- ◆ **個々のサーバ単位に、迅速な復元処理が可能。**ハードウェア障害からの復元には特に有用です。
- ◆ **高い拡張性。**数千万から数億単位のオブジェクトを保持した eDirectory データベースを持つサーバでもバックアップ可能です。バックアップの処理速度は主として I/O チャンネルの帯域幅で決まります。
- ◆ **レプリカ作成機能と DSMASTER サーバを組み合わせて運用していれば、ツリー全体の復元も容易。**DSMASTER サーバを設定していない場合でも、かなりの程度まで復元できます。[394 ページの「DSMASTER サーバによる災害対策」](#)を参照してください。
- ◆ **リモート操作が可能。**バックアップ / 復元処理のほとんどは、**iManager** を使って、Web ブラウザ画面から実行可能です。ファイアウォールの内側からでも外側からでも構いません。高度な処理は、コマンドラインから実行する Java クライアントである、**eMBox Client** を使い、リモート操作で実行できます。ファイアウォール越しにでも、あるいは VPN 経由でも操作できます。
- ◆ **関連ファイルもバックアップ可能。**データベース以外の関連ファイルも組にしてバックアップできます。NICI セキュリティファイル、ストリームファイル、インクルードファイルで指定したファイル (autoexec.ncf など) が対象になります。
- ◆ **サーバの停止直前の状態に eDirectory を復元可能。**ただしロールフォワードログを継続的に保存していることが条件です。[397 ページの「ロールフォワードログを使用する」](#)を参照してください。
- ◆ **ハードウェアのアップグレードを単純化。**eDirectory データベースのコールドバックアップを取り、新しいサーバに復元して、サーバの識別情報を簡単に新ハードウェアに移行できます。RAM のアップグレードなど、機器構成を変える際の安全措置としても有効です。[460 ページの「ハードウェアのアップグレードやサーバの交換」](#)を参照してください。

- ◆ **分散環境での運用を考慮。** ロールフォワードログがあれば、ツリー内の他のサーバと完全に同期した状態にまで復元できます。
- ◆ **無人でのバックアップが可能。** バッチファイルを作成して、eMBox クライアントを使用して無人でバックアップを実行できます。

新しい eDirectory バックアップツールである Backup eMTools には、個々のサーバ単位でデータベースおよび関連ファイルのバックアップを取り、復元するために必要な機能がすべて揃っています。個々のオブジェクトやツリーの一部を単位としてバックアップ / 復元することはできません。

システムバックアップ機能と組み合わせれば、eDirectory バックアップファイルをテープに保存して安全を期すことができます。

この章では、次のトピックについて説明します。

- ◆ [382 ページの「eDirectory のバックアップ処理に関する確認事項」](#)
- ◆ [385 ページの「バックアップサービスおよび復元サービスについて」](#)
- ◆ [397 ページの「ロールフォワードログを使用する」](#)
- ◆ [401 ページの「復元処理の準備」](#)
- ◆ [404 ページの「Novell iManager を使ったバックアップ / 復元作業」](#)
- ◆ [412 ページの「eMBox クライアントを使ったバックアップ / 復元作業」](#)
- ◆ [431 ページの「サーバ固有情報のバックアップに関する変更事項 \(NetWare のみ\)」](#)
- ◆ [432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)
- ◆ [437 ページの「バックアップ / 復元の運用例」](#)

## eDirectory のバックアップ処理に関する確認事項

**複数サーバ構成のツリーで、サーバが停止していてもオブジェクトにアクセスできるようにする**

- 複数サーバ構成のツリーでは、障害対策のため、すべての eDirectory パーティションについて、複数台のサーバにレプリカが作成されていることを確認します。

レプリカの作成については、[122 ページの「レプリカを追加する」](#)を参照してください。

**個々のサーバについて、ハードウェア障害などの場合に、迅速に完全復元できるようにするための準備**

- 定期的に(週1回など)eDirectoryデータベースのフルバックアップを取ってください。
- 定期的に(毎晩など)インクリメンタルバックアップを取ってください。
- eDirectory のフル / インクリメンタルバックアップ終了後、すぐにファイルシステムをテープにフル / インクリメンタルバックアップしてください。

Backup eMTool は、サーバ上の指定したディレクトリにバックアップファイルを作成しますが、これを直接テープに保存する機能はありません。したがって、eDirectory のバックアップ処理後すぐにファイルシステムのバックアップを行い、安全な記録媒体であるテープに保存する必要があります。

- 必要に応じて、ロールフォワードログを残すよう設定してください。

レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。他のサーバとレプリカを共有するデータベースは、停止直前の状態にまで復元しない限りデフォルトではオープンされません。

単一サーバ環境では、ロールフォワードログがなくても復元後の検証に失敗することはありませんが、最後にバックアップを取った時の状態にしか戻りません。ロールフォワードログがあれば、システム停止直前の状態まで復元できます。

ロールフォワードログ機能を使う際の主な注意点は次のとおりです。詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。

- ◆ ロールフォワードログの保存先を指定してください。初期設定を使用することはお勧めしません。

ログはサーバ上のローカルファイルとして保存する必要があります。障害対策上、eDirectory と同じディスクパーティション / ボリューム、同じ記憶デバイスは避けてください。ロールフォワードログ専用の、独立したパーティション / ボリュームを用意するとよいでしょう。

- ◆ ロールフォワードログの保存先を文書に記録しておき、障害時にはすぐにわかるようにしてください。

サーバが正常に動作していれば、iManager のバックアップ環境設定画面、または eMBox クライアントの getconfig オプションで調べることができます。ただしハードウェア障害などで eDirectory が使えない状態になると、この方法でロールフォワードログの場所を調べることはできません。

- ◆ ロールフォワードログを保存しているディスクパーティション / ボリュームの空き容量を監視し、容量不足にならないようにしてください。

容量不足のためロールフォワードログが作成できなくなると、eDirectory は応答しなくなります。

- ◆ ロールフォワードログの保存先にアクセスできるユーザを制限し、権利のないユーザがログを参照できないようにしてください。

- ◆ 復元が必要となったときは、復元処理の終了後、そのサーバのロールフォワードログ設定をやり直してください。復元処理の過程で、設定が初期状態に戻るためです。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

- NICI 機能を使用している場合は、NICI セキュリティファイルもバックアップの対象として設定してください。

このファイルがないと暗号化キーを復元できないため、暗号化されたデータを読むことができなくなります。NICI セキュリティの詳細については、『[NICI Administration Guide\(NICI 管理ガイド\)](#) (<http://www.novell.com/documentation/japanese/nici20/index.html>)』を参照してください。

- 複数サーバ構成のツリーの場合、バックアップ処理に Backup eMTool を使うためには、レプリカを共有するサーバすべてを eDirectory 8.5 以降にアップグレードする必要があります。

8.5 より前の eDirectory とは、復元後の検証処理の互換性がないためです。復元検証処理の詳細については、[388 ページの「Backup eMTool による復元作業の概要」](#)および [395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#)を参照してください。

- ❑ (NetWare® のみ)396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」を参照して、ファイルシステム上のアクセス権についても確認してください。起こりうる問題についてはあらかじめ確認し、必要に応じて予防措置を取ってください。
- ❑ 定期的にバックアップログを調べて、無人でのバックアップが正常に実行されていることを確認してください。
- ❑ サーバをアップグレードする際は、460 ページの「ハードウェアのアップグレードやサーバの交換」を参照してコールドバックアップを取ってください。
- ❑ 複数サーバ構成のツリーでは、障害対策のため、すべての eDirectory パーティションについて、複数台のサーバにレプリカが作成されていることを確認します。

パーティションのレプリカを作成しておけば、保守作業などのためサーバを停止している間もオブジェクトにアクセスできるばかりでなく、ハードウェア障害などでサーバが使えなくなった場合に備える障害対策としても役立ちます。逆に複数サーバ構成のツリーでレプリカを作成していないパーティションを保持するサーバがあると、障害時にパーティションを復元できない恐れがあります。したがって最善なのは、すべてのパーティションについてレプリカを作成しておくことです。複数サーバ構成のツリー内にレプリカを作成していないパーティションがある場合の問題点の詳細については、388 ページの「Backup eMTool による復元作業の概要」、397 ページの「ロールフォワードログを使用する」、および 432 ページの「復元後の検証処理に失敗した場合の対処方法」を参照してください。

レプリカの作成については、48 ページの「レプリカ」および 5 章 117 ページの、「パーティションおよびレプリカの管理」を参照してください。

- ❑ eDirectory および関連ファイルのバックアップを収めたテープは、安全な場所に保管するようにしてください。
- ❑ バックアップ計画が適切であるか、定期的に検証するようにしてください。
- ❑ (オプション)コールドバックアップ(データベースをクローズしてフルバックアップを取る作業)や高度なバックアップ/復元操作をリモート操作で実行する場合は、リモート側コンピュータに eMBox クライアントをインストールしてください。また、VPN を使用するなど、ファイアウォール越しにアクセスできるように設定する必要があります。

iManager を使用すればファイアウォールの外側からでも作業が可能ですが、コールドバックアップや高度な操作はサポートされていません。

eMBox クライアントは、eDirectory をインストールする際、同時にインストールされます。Sun JVM 1.3.1 が動作するワークステーション上でも使えます。eMBox クライアントのインストールや設定の手順については、479 ページの「eMBox コマンドラインクライアントの使用」を参照してください。

#### 災害により何台ものサーバが被害を受けた場合に備える準備

- ❑ 上述の対策はすべて実施してください。
- ❑ 複数サーバ構成のツリーであれば、障害対策のために DSMASTER サーバを用意するようお勧めします。  
394 ページの「DSMASTER サーバによる災害対策」を参照してください。
- ❑ 災害からの復旧計画が適切であるか、定期的に検証するようにしてください。

## バックアップサービスおよび復元サービスについて

- ◆ 385 ページの「eDirectory Backup eMTool について」
- ◆ 386 ページの「eDirectory 8.7.3 のバックアップ / 復元機能で変更された事項」
- ◆ 388 ページの「Backup eMTool による復元作業の概要」
- ◆ 389 ページの「バックアップファイルのヘッダ書式」
- ◆ 393 ページの「バックアップログファイルの書式」
- ◆ 394 ページの「DSMASTER サーバによる災害対策」
- ◆ 395 ページの「遷移ベクトルと復元後の検証処理」
- ◆ 395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」
- ◆ 396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」

### eDirectory Backup eMTool について

Backup eMTool には、個々のサーバ単位で、稼動したままの状態継続的に eDirectory データベースのバックアップを取る機能があります。データベースを停止することなく、処理を始めた時点のバックアップを取ることができます。つまり、バックアップ作業はいつでも可能であり、その間も eDirectory を使い続けることができます (特に指示しなければこの「ホット」バックアップになります。必要であればデータベースを停止して「コールド」バックアップを取ることも可能です)。

また、ロールフォワードログを有効にすれば、最後にバックアップを取った時点以降のトランザクションをすべて記録しておけます。これを使えば、サーバが停止する直前の状態に復元することもできます。さらに、レプリカリングに属するサーバすべてについてロールフォワードログを取る必要があります。これにより、他のサーバとの同期状態も復元できます。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。ロールフォワードログの機能は、デフォルトでは無効になっています。詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。

新しいバックアップツールである Backup eMTool も、eDirectory のオブジェクトをすべて一度にバックアップできるわけではありません。各サーバのパーティションが単位となります。この方式は、特定のサーバのみを復元する場合に優れているばかりでなく、TSA for NDS<sup>®</sup> を使う旧式の方法に比べて高速です。ただし、eDirectory 8.6 のマニュアルにも記載されているように、TSA for NDS は今でも使用できます。必要に応じて TSA for NDS と Backup eMTool を使い分けることができます。動作の違いについては、[386 ページの「eDirectory 8.7.3 のバックアップ / 復元機能で変更された事項」](#)を参照してください。

eDirectory の新しいバックアップツールは、システムバックアップ機能と組み合わせ使用して、eDirectory バックアップファイルをテープに安全に保存する必要があります。Novell はバックアップ用製品を開発している主要企業と提携しています。具体的な社名については、[Backup Partners \(バックアップ関連の提携企業\)](http://www.novell.com/products/netware/partners/backup.html) (<http://www.novell.com/products/netware/partners/backup.html>) を参照してください。

NetWare の場合、このバックアップツールを使うためには、ファイルシステムに対する適切なアクセス権が必要です。詳細については、[396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」](#)を参照してください。

iManager からは、コールドバックアップ、無人バックアップ、高度な復元機能を除くバックアップ / 復元機能を実行できます。詳しくは [404 ページの「Novell iManager を使ったバックアップ / 復元作業」](#) を参照してください。一方、eMBox Java コマンドラインクライアントからは、無人バックアップを含め、すべてのバックアップ / 復元作業が実行できます。詳しくは [412 ページの「eMBox クライアントを使ったバックアップ / 復元作業」](#) を参照してください。

iManager から実行できるバックアップ / 復元オプションについては、オンラインヘルプを参照してください。eMBox クライアントから実行できる機能については、[423 ページの「バックアップ / 復元のコマンドラインオプション」](#) を参照してください。

復元の具体的な処理過程については、[388 ページの「Backup eMTool による復元作業の概要」](#) を参照してください。

eDirectory Backup eMTool は、eMBox ツールのひとつです。eMBox は、eDirectory を構成する一部としてサーバにインストールされるサービスです。

Backup eMTool は次のファイルから成ります。

ファイル名	説明
backupcr	バックアップ / 復元機能をすべて含むコアライブラリ。  ユーザインタフェースはなく、backuptl プログラムから動的にロード、リンクされる形で動作します。
backuptl	backupcr ライブラリを呼び出す eMTool のインタフェース部分。eMBox アーキテクチャを介してバックアップ / 復元機能を提供するプログラムです。  iManager プラグイン、eMBox クライアント、Java コマンドラインクライアントを介して使用できます。
dsbackup_en.xlf	Backup eMTool で表示されるメッセージの言語ファイルです。

Backup eMTool で作成されるバックアップファイルやログファイルの形式については、[393 ページの「バックアップログファイルの書式」](#) および [389 ページの「バックアップファイルのヘッダ書式」](#) を参照してください。

**重要：** 復元後の検証処理については、8.5 より前の eDirectory とは互換性がありません。レプリカリングに属するサーバで新しいバックアップ / 復元ツールを使う場合は、これに属するサーバすべてを eDirectory 8.5 以降にアップグレードする必要があります ([395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#) も参照してください)。

## eDirectory 8.7.3 のバックアップ / 復元機能で変更された事項

eDirectory の以前のバージョンでは、バックアップ / 復元ツールは、ツリーをオブジェクト単位でバックアップする方式を使用していました。

eDirectory 8.7 で組み込まれた Backup eMTool は、方式やアーキテクチャが一新されています。すなわち、ツリーではなくサーバを単位とし、個々のサーバの eDirectory データベースをバックアップする、という方式になったのです。この変更により、以前の TSA for NDS よりもバックアップ処理速度が大幅に改善されました。

TSA for NDS は今でも使用できますが、今後は新しいバックアップツールを使うようお勧めします。

サーバ固有の情報も Backup eMTool でバックアップできます。431 ページの「サーバ固有情報のバックアップに関する変更事項 (NetWare のみ)」を参照してください。

新旧バックアップツールの違いを次の表に示します。

項目	TSA for NDS による以前のバックアップ	Backup eMTool による「ホットバックアップ」
バックアップの対象	ツリー全体をオブジェクト単位でバックアップ。  旧バックアップツール(必要ならば8.7でも利用可能)の詳細については、『Novell eDirectory 8.6 管理ガイド』の「Novell eDirectory のバックアップと復元 ( <a href="http://www.novell.com/documentation/japanese/ndsedir86/taoenu/data/a2n4mb6.html">http://www.novell.com/documentation/japanese/ndsedir86/taoenu/data/a2n4mb6.html</a> )」を参照してください。	サーバ単位で eDirectory データベースをバックアップ。  ツリー全体の耐障害性は主としてレプリカ作成機能で確保していますが、サーバ単位のバックアップ機能によりさらに強固になります。  災害などで何台ものサーバが停止した場合でもツリーを復元できるようにするためには、DSMASTER サーバを導入し、それに応じたレプリカ作成方針を立てる必要があります。394 ページの「DSMASTER サーバによる災害対策」を参照してください。
処理速度	-	大幅に改善されています。新規バックアップツールの開発に当たっては、処理性能を最重視しました。
バックアップファイルの保存先	直接テープに書き出し可。	ファイルシステム上のバックアップファイルとして書き出し。  別途ファイルシステムのバックアップツールを使って、テープに書き出す必要があります。
プラットフォーム間の違い	プラットフォームによって使い方が別々。	どのプラットフォームでも同じ使い方。
個々のサーバ単位での復元	考慮されていません。	サーバ単位での復元処理が可能。ハードディスク障害時のほか、サーバ機を移行する際にも利用できます。  ロールフォワードログを使えば、停止直前の状態に復元することも可能です。これにより、レプリカリングに属する他のサーバと同期状態を揃えることができます。  eDirectory の関連ファイルもバックアップの対象とすることができます。たとえば NICI ファイルのバックアップと復元ができます。独自の関連ファイルリストを作成して、対象ファイルをバックアップに追加することもできます。
NICI ファイルのバックアップ / 復元	考慮されていません。	NICI ファイルのバックアップ / 復元も可能です。これにより、復元後、暗号化データにアクセスできます。復元作業の時間を大幅に短縮できるでしょう。

項目	TSA for NDS による以前のバックアップ	Backup eMTool による「ホットバックアップ」
個々のサーバのロールフォワードログ	考慮されていません。	最後にバックアップを取った時点以降のトランザクションを、ロールフォワードログとしてすべて記録しておけます。これを使うと、停止直前の状態にまで復元することができます。複数サーバ環境では、これを使用して他のサーバと同期状態を揃えることができます。ロールフォワードログの機能は、デフォルトでは無効になっています。詳細については、 <a href="#">397 ページの「ロールフォワードログを使用する」</a> を参照してください。

## Backup eMTool による復元作業の概要

復元作業に先立ち、バックアップファイルをすべて揃えておく必要があります。その手順については [401 ページの「復元処理の準備」](#)を参照してください。iManager や eMBox クライアントから Backup eMTool の復元機能呼び出すと、Backup eMTool で次の処理が行われます。

1. DS エージェントをクローズします。
2. アクティブな DIB(Data Information Base) セットを、NDS から RST に切り替えます。  
既存の NDS データベースはそのままサーバに残り、復元後の検証に失敗した場合は、再びこれがアクティブな DIB セットになります。
3. 復元処理が始まります。新たに RST という DIB セットを作り、そこに復元します。
4. DIB セットをいったん無効にします。

擬似サーバのログイン無効属性をオンにします。これは、この DIB セットを使って DS エージェントがオープンされるのを避けるための措置です。

5. ロールフォワードログに関する設定をデフォルトに戻します。

したがって、復元後はロールフォワードログを書き出さない設定になります。書き出し先ファイル名の設定もデフォルトに戻ります。

このサーバでロールフォワードログ機能を使うためには、復元後に改めて有効に切り替え、障害対策のための書き出し先も設定し直して、ロールフォワードログの環境設定を再作成する必要があります。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

6. 復元された RST データベースの検証処理を行います。

復元されたデータの整合性を確認します。この確認は、レプリカを共有しているすべてのサーバにアクセスし、遷移ベクトルを比較して実行されます。

検証結果はログファイルに出力されます。

リモートサーバの遷移ベクトルの方がローカルベクトルより後の時間に作成されたものである場合は、復元されなかったデータがあるということなので、検証処理は失敗します。

あるレプリカの検証処理に失敗した場合、ログにはたとえば次のよう出力されず。遷移ベクトルを比較している様子がわかります。



```

Server:\T=LONE_RANGER\O=novell\CN=CHIP
Replica:\T=LONE_RANGER\O=novell
Status:ERROR = -6034
Local TV Remote TV
s3D35F377 r02 e002 s3D35F3C4 r02 e002
s3D35F370 r01 e001 s3D35F370 r01 e001
s3D35F363 r03 e001 s3D35F363 r03 e001
s3D35F31E r04 e004 s3D35F372 r04 e002
s3D35F2EE r05 e001 s3D35F2EE r05 e001
s3D35F365 r06 e003 s3D35F365 r06 e003

```

詳細については、[395 ページの「遷移ベクトルと復元後の検証処理」](#)を参照してください。

7. 検証に成功した場合は、RST を NDS と改名し、ログイン無効属性をオフにします。したがってこれがサーバでアクティブな eDirectory データベースになります。失敗した場合は改名しないので、元の NDS が再びアクティブな DIB セットになります。

検証に失敗した場合の復旧方法については、[432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)を参照してください。「高度な復元オプション」を使えば、強制的に RST データベースをアクティブにし、ロックを解除することもできますが、Novell の担当者の指示がない場合はお勧めできません。

## バックアップファイルのヘッダ書式

バックアップファイルのヘッダには、次のような重要な情報が記録されています。

- ◆ バックアップファイルが作成された時点のファイル名。  
バックアップ作成後にファイル名を変更した場合にはこの情報が有用になります。
- ◆ バックアップ時点のロールフォワードログ名。  
フルバックアップと 3 回分のインクリメンタルバックアップというように複数のファイルがある場合、最後のバックアップファイルに記載されたロールフォワードログ名が重要です。バックアップファイルからの復元処理後、どのロールフォワードログから適用するかを確認するために使用されるからです。
- ◆ このサーバが保持しているレプリカのリスト。  
各レプリカをどのサーバに配置しているか記録していない場合に役立ちます。災害で何台ものサーバが停止した場合、バックアップファイルのヘッダに表示されたこの情報を見れば、最初にどのサーバを復元するべきか判断できます。
- ◆ 同時にバックアップするようユーザのインクルードファイルに指定された関連ファイル名。
- ◆ 複数のファイルに分割してバックアップする場合のファイル数。

各バックアップファイルのヘッダは XML 形式で記述されます。ヘッダ部に続き、データベース内のデータを、バイナリ形式で記録します (ファイルの末尾にバイナリデータがあると構文解析でエラーが発生しますが、XML ヘッダは XML 標準に適合しています)。バックアップが複数のファイルに分かれる場合、各ファイルに同じヘッダ情報を記録します。

**警告** : バックアップファイルを開く場合でもヘッダを確認するだけにとどめ、保存や変更はしないようにしてください。ファイルの一部が切り捨てられてしまうことがあります。ほとんどのアプリケーションではバイナリデータを正確に保存することはできません。

XML ヘッダの DTD を次に示します (この DTD は参照のため、バックアップファイルのヘッダの一部として書き込まれます)。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE backup [
<!ELEMENT backup (file|replica)*>
<!ELEMENT file (#PCDATA)>
<!ELEMENT replica EMPTY>
<!ATTLIST backup version CDATA #REQUIRED
  backup_type (full|incremental) #REQUIRED
  idtag CDATA #REQUIRED
  time CDATA #REQUIRED
  srvname CDATA #REQUIRED
  dsversion CDATA #REQUIRED
  compression CDATA "none"
  os CDATA #REQUIRED
  current_log CDATA #REQUIRED
  number_of_files CDATA #IMPLIED
  backup_file CDATA #REQUIRED
  incremental_file_ID CDATA #IMPLIED
  next_inc_file_ID CDATA #IMPLIED>
<!ATTLIST file size CDATA #REQUIRED
  name CDATA #REQUIRED
  encoding CDATA "base64"
  type (user|nici) #REQUIRED>
<!ATTLIST replica partition_DN CDATA #REQUIRED
  modification_time CDATA #REQUIRED
  replica_type (MASTER|SECONDARY|READONLY|SUBREF|
  SPARSE_WRITE|SPARSE_READ|Unknown) #REQUIRED
  replica_state (ON|NEW_REPLICA|DYING_REPLICA|LOCKED|
  CRT_0|CRT_1|TRANSITION_ON|DEAD_REPLICA|
  BEGIN_ADD|MASTER_START|MASTER_DONE|
  FEDERATED|SS_0|SS_1|JS_0|JS_1|MS_0|MS_1|
  Unknown) #REQUIRED>
]>
```

DTD に含まれる属性について次の表で説明します。

属性	説明
backup version	バックアップツールのバージョン。
backup backup_type	フルバックアップかインクリメンタルバックアップかの別(コールドバックアップはフルバックアップとして扱います)。
backup idtag	バックアップ時刻に基づいて生成された GUID。バックアップファイル名が変わっていても、これを使えばバックアップを識別できます。
backup time	バックアップ処理の開始日時。
backup srvname	バックアップ対象サーバの識別名。
backup dsversion	サーバ上で稼動している eDirectory のバージョン。
backup compression	Backup eMTool がバックアップデータを圧縮したかどうか。対象になるのはバックアップデータ本体だけで、ヘッダは圧縮されません。
backup os	バックアップ処理を実行したオペレーティングシステム。復元はこれと同じオペレーティングシステムのみで行うようお勧めします。

属性	説明
backup current_log	復元に必要な最初のロールフォワードログ。正しく復元するために必要な情報です。
backup number_of_files	分割してバックアップする場合のファイル数。ひとつ目のバックアップファイルにのみ記録されます。
backup backup_file	このバックアップファイル名。  複数のファイルに分けてバックアップする場合、各ファイル名には順序番号が付きませんが、それも含むファイル名が記録されます。複数のバックアップファイルのファイル名の例は、「-s <i>file_size</i> 」を参照してください。
backup incremental_file_ID	インクリメンタルバックアップの場合、そのファイルの ID。
backup next_inc_file_ID	これに続くインクリメンタルバックアップに与える ID。正しく復元するために必要な情報です。
file size	このファイルの <file> タグ間にあるデータ長。
file name	バックアップファイル名およびその保存先。
file encoding	ファイルのエンコードに使ったアルゴリズム。
file type	これが NCI ファイルか、それ以外にユーザが指定したファイルかの別。
replica partition_DN	パーティションの識別名。  各レプリカをどのサーバに配置しているか記録していない場合に役立ちます。災害で何台ものサーバが停止した場合、バックアップファイルのヘッダに表示されたこの情報を見れば、最初にどのサーバを復元すべきか判断できます。
replica modification_time	バックアップ時点の、このレプリカの遷移ベクトル。
replica replica_type	レプリカの種別。マスタ、読み込み専用など。
replica_state	バックアップ時点でのレプリカの状態。「On」、「New Replica」など。

バックアップファイルのヘッダ例を次に示します。これは Windows NT サーバで作成したもので、NCI セキュリティファイルもバックアップ対象になっています。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE backup [
<!ELEMENT backup (file|replica)*>
<!ELEMENT file (#PCDATA)>
<!ELEMENT replica EMPTY>
<!--ATTLIST backup version CDATA #REQUIRED
      backup_type (full|incremental) #REQUIRED
      idtag CDATA #REQUIRED
      time CDATA #REQUIRED
      srvname CDATA #REQUIRED
      dsversion CDATA #REQUIRED
      compression CDATA "none"
      os CDATA #REQUIRED
      current_log CDATA #REQUIRED
      number_of_files CDATA #IMPLIED
      backup_file CDATA #REQUIRED
      incremental_file_ID CDATA #IMPLIED
```

```

    next_inc_file_ID CDATA #IMPLIED>
<!ATTLIST file size CDATA #REQUIRED
    name CDATA #REQUIRED
    encoding CDATA "base64"
    type (user|nici) #REQUIRED>
<!ATTLIST replica partition_DN CDATA #REQUIRED
    modification_time CDATA #REQUIRED
    replica_type (MASTER|SECONDARY|READONLY|SUBREF|
    SPARSE_WRITE|SPARSE_READ|Unknown) #REQUIRED
    replica_state (ON|NEW_REPLICA|DYING_REPLICA|LOCKED|
    CRT_0|CRT_1|TRANSITION_ON|DEAD_REPLICA|
    BEGIN_ADD|MASTER_START|MASTER_DONE|
    FEDERATED|SS_0|SS_1|JS_0|JS_1|MS_0|MS_1|
    Unknown) #REQUIRED>
]>

<backup version="2" backup_type="full" idtag="3D611DA2" time="2002-8-
19'T10:32:35" srvname="\T=MY_TREE\O=novell\CN=DSUTIL-DELL-NDS"
dsversion="1041081" compression="none" os="windows"
current_log="00000003.log" next_inc_file_ID="2" number_of_files="0000001"
backup_file="c:\backup\header.bak">

<replica partition_DN="\T=MY_TREE" modification_time="s3D611D95_r1_e2"
replica_type="MASTER" replica_state="ON" />

<replica partition_DN="\T=MY_TREE\O=part1"
modification_time="s3D611D95_r1_e2" replica_type="MASTER" replica_state="ON"
/>

<replica partition_DN="\T=MY_TREE\O=part2"
modification_time="s3D611D95_r1_e2" replica_type="MASTER" replica_state="ON"
/>

<replica partition_DN="\T=MY_TREE\O=part3"
modification_time="s3D611D96_r1_e2" replica_type="MASTER" replica_state="ON"
/>

<file size="190" name="C:\WINNT\system32\novell\nici\bhawkins\XARCHIVE.001"
encoding="base64" type="nici">the data is included here</file>

<file size="4228" name="C:\WINNT\system32\novell\nici\bhawkins\XMGRCFG.KS2"
encoding="base64" type="nici">the data is included here</file>

<file size="168" name="C:\WINNT\system32\novell\nici\bhawkins\XMGRCFG.KS3"
encoding="base64" type="nici">the data is included here</file>

<file size="aac" name="C:\WINNT\system32\novell\nici\nicintacl.exe"
encoding="base64" type="nici">the data is included here</file>

<file size="150" name="C:\WINNT\system32\novell\nici\nICISDI.KEY"
encoding="base64" type="nici">the data is included here
</file>

<file size="4228" name="C:\WINNT\system32\novell\nici\system\Xmgrcfg.ks2"
encoding="base64" type="nici">the data is included here
</file>

<file size="168" name="C:\WINNT\system32\novell\nici\system\Xmgrcfg.ks3"
encoding="base64" type="nici">the data is included here
</file>

<file size="1414" name="C:\WINNT\system32\novell\nici\xmgrcfg.wks"
encoding="base64" type="nici">the data is included here

```

</file>

</backup>

ヘッダ部に続き、データベースのバックアップデータをバイナリ形式で格納します。

## バックアップログファイルの書式

eDirectory Backup eMTool は、前回までのバックアップを含め、処理内容を細かく記録したログを残すようになっています。このログファイルにはすべてのバックアップ履歴、バックアップの開始および終了時刻、およびバックアッププロセス中に発生した考えられるエラーについての情報が含まれます。前回までのログに追記する形で記録します。書き出し先を別途指定することもできます。

無人バックアップが正常に実行されているか確認するためにも、このログは重要です。最終行を見ると、成功か失敗かの区別およびエラーコードがわかります。

Backup eMTool のログファイルには、過去のバックアップ処理の ID も記録されています。これは復元に必要なフルバックアップ、インクリメンタルバックアップのファイルを間違いなく揃えるために役立ちます。先頭の 4 行は、バックアップファイルのヘッダ情報をそのまま複写したものです。

また、同時にバックアップしたファイル名も記録します。NICI ファイルや、インクルードファイルでユーザが指定したファイルがこれに当たります。

復元処理の際は、実際に復元されたファイルを記録します。

ログファイルの出力例を次に 2 つ示します。

```
|=====DSBackup Log:Backup=====|
Backup type:Full
Log file name:sys:/backup/backup.log
Backup started:2002-6-21'T19:53:5GMT
Backup file name:sys:/backup/backup.bak
Server name:\T=VIRTUALNW_TREE\O=novell\CN=VIRTUALNW
Current Roll Forward Log:00000001.log
DS Version:1041072
Backup ID:3D138421
Backing up security file:sys:/system/nici/INITNICI.LOG
Backing up security file:sys:/system/nici/NICISDI.KEY
Backing up security file:sys:/system/nici/XARCHIVE.000
Backing up security file:sys:/system/nici/XARCHIVE.001
Backing up security file:sys:/system/nici/XMGRCFG.KS2
Backing up security file:sys:/system/nici/XMGRCFG.KS3
Backing up security file:sys:/system/nici/XMGRCFG.NIF
Starting database backup...
Database backup finished
Completion time 00:00:03
Backup completed successfully
```

```
|=====DSBackup Log:Restore=====|
Log file name:sys:/save/doc.log
Restore started:2002-7-19'T19:1:34GMT
Restore file name:sys:/backup/backup.bak
Starting database restore...
Restoring file sys:/backup/backup.bak
Restoring file sys:/system/nici/INITNICI.LOG
Restoring file sys:/system/nici/NICISDI.KEY
Restoring file sys:/system/nici/XARCHIVE.000
```

```
Restoring file sys:/system/nici/XARCHIVE.001
Restoring file sys:/system/nici/XMGRCFG.KS2
Restoring file sys:/system/nici/XMGRCFG.KS3
Restoring file sys:/system/nici/XMGRCFG.NIF
Database restore finished
Completion time 00:00:15
Restore completed successfully
```

## DSMASTER サーバによる災害対策

複数サーバ環境で、サーバがすべて失われてしまうような災害にも備えるためには、ツリーの一部として DSMASTER サーバを導入することを検討してください。

Backup eMTool は各サーバを個別にバックアップするツールです。すなわち、ツリー全体ではなく、個々のサーバを対象とするよう設計されています。ただし DSMASTER サーバを作成しておけば、Backup eMTool でも、ツリー構造全体をバックアップできるようになります。運用例の概要については、[441 ページの「シナリオ：複数サーバ構成のネットワークで、すべてのサーバが使えなくなった場合」](#)を参照してください。

災害からの復旧で問題となるのは、同じパーティションのレプリカが複数あって、互いに整合性が取れていない場合の対処方法です。災害によってロールフォワードログも失われている場合、すべてのサーバを同じ時点の状態に復元することはできません。バックアップされたレプリカはサーバごとに異なる時点のものなので、ロールフォワードログを使わずに単純に復元しただけでは、全体をツリーとしてまとめる際に問題が起こるためです。復元後の検証処理は、このような問題を予防するために設計されています。デフォルトでは、他のレプリカとの不整合が見つければ、eDirectory データベースは復元後にはオープンされません。

DSMASTER サーバを導入すればこのような事態に備えることができます。ツリー全体のマスタコピーを作成しておき、これを復元の基準点として使用します。

DSMASTER サーバを導入する手順を次に示します。

- ◆ ある 1 台のサーバに、ツリーに属するパーティションすべてのレプリカを保持するよう設定してください。これにより、ツリー全体のコピーが、特定のサーバの eDirectory データベース中に作られるようになります。ただし、ツリーの容量が大きい場合は、複数台のキーサーバに分けても構いません。これを DSMASTER サーバと呼びます。DSMASTER サーバに作るレプリカは、マスタまたは読み書き用と設定してください。

**注：**複数台のキー DSMASTER サーバに分割する場合は、同じパーティションのレプリカが 2 台以上に作られることのないようにしてください。重複のない構成にしておけば、災害後の復元作業において、レプリカ間に不整合が生じることはありません。

災害から復旧する際に、最新のロールフォワードログは使えません。このログは各サーバのローカルファイルとして保存されるためです。したがって、複数台の DSMASTER サーバを、すべて同じ時点の状態に戻すことはできない場合があります。2 台の DSMASTER サーバに同じレプリカがあれば、これが一致せず、ツリーに不整合が生じる恐れがあります。そのため、災害対策としては、同じパーティションのレプリカを複数の DSMASTER サーバに作成することは避ける必要があります。

レプリカ全般については、[48 ページの「レプリカ」](#)を参照してください。

- ◆ DSMASTER サーバを定期的にバックアップして、ツリー全体のバックアップコピーを作成してください。災害対策としては、DSMASTER サーバのバックアップ以外にも、十分な予防措置を講じておくといよいでしょう。

以上のようにツリーを設計しておくことで、災害が起こっても、迅速にツリー構造を再構築し、稼働させることができます。1 台だけのサーバ（あるいは少数のキーサーバ）のみを復元し、これをマスタレプリカとして扱うようにすればよいのです。

このツリーが稼動し始めてから、フル/インクリメンタルバックアップファイルを使用して、DSMASTER 以外のサーバも順次復元していきます。ロールフォワードログがないため、他のサーバに対する復元後の検証処理は失敗します。そこで、レプリカリングからいったん外し、DSRepair を使って、すべてのレプリカ情報を外部参照に変更します。その後、DSMASTER サーバ上のコピーからレプリカを作成して、改めてサーバにレプリカを追加します。この手順については、[432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)に記載されています。

災害により一部のサーバが失われた場合、操作手順はやや複雑になりますので、Novell の担当者に連絡してください。

## 遷移ベクトルと復元後の検証処理

遷移ベクトルとは、レプリカのタイムスタンプのことです。レプリカ作成時刻を 1970 年 1 月 1 日からの経過秒数で表したものと、レプリカ番号、および現在のイベント番号を組にして表示されます。たとえば次のような形をしています。

```
s3D35F377 r02 e002
```

バックアップ/復元処理に関していえば、復元されたサーバがレプリカリング内で正しく同期しているかどうか確認するために使う、重要なデータです。

同じパーティションのレプリカを保持するサーバは、レプリカの同期を取るため、互いにデータをやり取りしています。サーバはレプリカリング内の他のサーバと通信するたびに、他のサーバの遷移ベクトルを記録しています。遷移ベクトルを使用すると、レプリカリング内の各レプリカが同期を保つためには、どのデータを送ればよいか、サーバが常に把握できます。サーバが停止するとこの通信が止まり、再び通信できるようになるまでの間、遷移ベクトルに更新や変更があっても他のサーバはそれを送信しません。

あるサーバの eDirectory を復元した後、検証処理として、復元された遷移ベクトルをレプリカリングに属する他のサーバと比較します。これは、復元されたレプリカが他のサーバと同期の取れた状態であるかどうか確認するために実行されます。

リモートサーバの遷移ベクトルの方がローカルベクトルより後の時間に作成されたものである場合は、復元されなかったデータがあるということなので、検証処理は失敗します。これは、フルバックアップまたはインクリメンタルバックアップの前に、ロールフォワードログの機能を有効にしていなかった、ロールフォワードログを使わずに復元しようとした、または必要なロールフォワードログが揃っていなかった、などが原因でデータの損失があることが考えられます。

デフォルトでは、復元した eDirectory データベースが他のレプリカと整合が取れていない場合、そのままではオープンできません。

遷移ベクトルに不整合があるログファイルの例については、[388 ページの「Backup eMTool による復元作業の概要」](#)を参照してください。

互換性の問題で検証処理に失敗することもあります。これについては、[395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#)を参照してください。

検証に失敗した場合の対処方法については、[432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)を参照してください。

## 復元後の検証については eDirectory 8.5 以降のみで互換性がある

復元後の検証処理については、8.5 より前の eDirectory とは互換性がありません。レプリカリング内に eDirectory 8.5 より前のバージョンが稼動しているサーバがあれば、復元処理は失敗します。エラーコードは -666、すなわち「DS バージョンの不整合」となりま

す。これは、レプリカが同期していないことを示すのではなく、eDirectory のバージョンが 8.5 以前のため、遷移ベクトルの比較ができなかったことを示しているに過ぎません。

デフォルトでは、検証処理に失敗しているため、データベースがオープンされません。ただし、エラーが発生しているのが 8.5 サーバのみで、他のサーバでは問題なく検証されているのであれば、eMBox クライアントで上書き復元を実行することにより、安全にデータベースをオープンすることができます。

あるいは、旧バージョンのサーバをレプリカリングから外し、再度復元を試みる方法もあります。

復元処理と遷移ベクトルの詳細については、[388 ページの「Backup eMTool! による復元作業の概要」](#) および [395 ページの「遷移ベクトルと復元後の検証処理」](#) を参照してください。

## NetWare のファイルシステムデータを復元する際のアクセス権の保存

NetWare の場合、ファイルシステム権 (トラスティの割り当て) は、eDirectory にあるトラスティオブジェクトによって決まります。そのため、eDirectory と NetWare のファイルシステムデータを復元する際には、ファイルシステム権について注意する必要があります。

eDirectory の後でファイルシステムデータを復元するようになれば、その時点でファイルシステム権も正しく復元されます。ただしこの問題を認識しておくことは必要です。起こりうる問題についてはあらかじめ確認し、必要に応じて予防措置を取ってください。

### 復元処理によりファイルシステム権に影響が及ぶ理由

eDirectory の復元作業に先立ち、eDirectory を新たにインストールし、仮のツリーを作成しておく必要があります。仮のツリーを作成するのは故障した記憶デバイスに代わる新しいデバイス上でも、サーバを移行する場合であれば新しいコンピューター上でもかまいません。

eDirectory を新規インストールした段階では、トラスティ権が割り当てられたオブジェクトはありません (復元処理の過程で、トラスティオブジェクトも復元されることになります)。

ファイルシステムデータを復元する際、eDirectory 内のトラスティオブジェクトが検索されます。しかし、復元前に新規インストールした状態のままトラスティオブジェクトが見つからない場合、そのオブジェクトに対する権利の割り当てもファイルシステムから削除されることがあります。

### この問題への対処方法

復元とファイルシステム権利 / トラスティ割り当てに関する問題には、次のように、少しずつ異なるいくつかの対処方法があります。

- ◆ 最も重要なのは、eDirectory を先に復元し、その後でファイルシステムを復元することです。

eDirectory を新たにインストールし、復元する作業に、特別な準備はいりません。これが済んでから、ファイルシステムの復元ツールを使って、ファイルシステム権利やトラスティ割り当てを元に戻すために必要なファイルを復元できます。

- ◆ バックアップの段階で、trustbar.nlm を使用してファイルシステム権利 / トラスティ割り当てをバックアップできます。同等の処理ができるサードパーティ製ソフトウェア



アを使っても構いません。こうしておけば、eDirectory を復元した後でも、必要に応じてトラスティ割り当てを復元できます。

ファイルシステム権利 / トラスティ割り当てのバックアップは、eDirectory やファイルシステムと同じスケジュールで実施することもできます。

**注:** ファイルシステム権利のバックアップのスケジュールリングは、サードパーティ製ソフトウェアのほか、Novell Support Web サイトで提供している [cron.nlm \(http://support.novell.com/servlet/tidfinder/2939440\)](http://support.novell.com/servlet/tidfinder/2939440) でも可能です。

- ◆ ストレージシステムの構成を検討し直すことにより、eDirectory やファイルシステムデータを復元しなければならないような障害の可能性を減らすことができます。たとえば RAID 構成などにすれば、ディスクが 1 台故障した場合にデータを損失する可能性は低くなります。sys: ボリュームを冗長構成にしておけば、デバイスに障害があっても、eDirectory の新規インストールやファイルシステムの復元作業はしないで済むかもしれません。
- ◆ 何らかの理由により、先にファイルシステムデータを復元してしまい、権利が失われてしまったとしても、eDirectory を復元した後、もう一度ファイルシステムを復元すれば元に戻すことができます。
- ◆ eDirectory の復元作業が終わるまで、sys: 以外のボリュームはマウントしないでおく、という方法もあります。障害を受けたのは sys: ボリュームだけで、ほかは動作している、という場合に有効です。

確実にボリュームがマウントされないようにするためには、その記憶デバイスとサーバを結ぶケーブルを物理的に外してから NetWare や eDirectory をインストールし、復元作業の終了後、再び接続し直すようにするとよいでしょう。

eDirectory の復元後、必要に応じて sys: ボリュームのファイルシステムを復元して、権利の設定を復旧してください。

## ロールフォワードログを使用する

ロールフォワードログとは、他のデータベース製品でいう「ジャーナル」に相当する機能です。ロールフォワードログ (RFL) は、データベースの変更をすべて記録したものです。

ロールフォワードログを使用する利点は、最後のフル / インクリメンタルバックアップ以降の変更履歴が得られるため、障害で停止する直前の状態にまで eDirectory を復元できることです。ロールフォワードログを使用しないと、最後のフル / インクリメンタルバックアップを取った時点までしか eDirectory を復元できません。

eDirectory は、トランザクションをデータベースに反映する前に、その操作内容をログファイルに記録するようになっています。デフォルトでは、ログファイルはディスク容量の節約のために次々に重ね書きされるようになっているため、eDirectory の変更履歴は残りません。

継続的にロールフォワードログを取る設定にすると、変更履歴が連続したロールフォワードログファイルに保存されます。ロールフォワードログはサーバの性能には影響がありません。eDirectory がすでに作成しているログファイルのエントリを単に保存するだけです。

レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。他のサーバとレプリカを共有するデータベースは、停止直前の状態にまで復元しない限りデフォルトではオープンされません。ロールフォワードログがない場合は、[432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)で説明する手順で復旧してください。

ロールフォワードログの機能は、デフォルトでは無効になっています。サーバで必要に応じて有効に切り替えてください。ロールフォワードログは、サーバの復元作業を実行すると再び無効になり、設定がデフォルトに戻ります。このため復元後に再び有効にし、設定を再作成した上で、改めてフルバックアップを取ってください。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

単一サーバ環境ではロールフォワードログがなくても構いません。しかしロールフォワードログがあれば、最後のバックアップ時ではなくシステム停止直前の状態に復元できます。

ロールフォワードログをオンにする場合、ディスクの空き容量は常に監視している必要があります。詳細については、[400 ページの「ロールフォワードログのバックアップと削除」](#)を参照してください。

このセクションでは、次のトピックについて説明します。

- ◆ [398 ページの「ロールフォワードログ機能を使用する上での注意事項」](#)
- ◆ [399 ページの「ロールフォワードログの保存先」](#)
- ◆ [400 ページの「ロールフォワードログのバックアップと削除」](#)
- ◆ [401 ページの「注意：eDirectory を削除するとロールフォワードログも削除される問題」](#)

ロールフォワードログ機能の切り替えや設定には、iManager または eMBox クライアントを使います。[407 ページの「iManager によるロールフォワードログの設定」](#)または [418 ページの「eMBox クライアントによるロールフォワードログの設定」](#)を参照してください。

## ロールフォワードログ機能を使用する上での注意事項

継続的にロールフォワードログ機能を使用する場合、次の点に注意してください。

- ◆ バックアップ処理の実行前にロールフォワードログ機能を有効にしておかないと、データベースの復元に利用することはできません。
- ◆ 障害に備えるため、eDirectory とは別の記憶デバイスにロールフォワードログを保存するようにしてください。セキュリティを考慮すれば、ログへのアクセス権も制限する必要があります。詳細については、[399 ページの「ロールフォワードログの保存先」](#)を参照してください。
- ◆ ロールフォワードログの保存先を文書に記録しておいてください。詳細については、[399 ページの「ロールフォワードログの保存先」](#)を参照してください。
- ◆ ログの保存先のディスクの空き容量を常に監視している必要があります。詳細については、[400 ページの「ロールフォワードログのバックアップと削除」](#)を参照してください。
- ◆ ロールフォワードログ機能が無効になっていたり、ログファイルを損失した場合は、有効に切り替えた後、改めてフルバックアップを取ってください。そうしなければ完全に復元できなくなる恐れがあります。次のような状況の場合に必要です。
  - ◆ 復元処理の直後。復元処理の過程で、ロールフォワードログ機能は無効になり、設定もデフォルト値に戻ってしまいます。
  - ◆ デバイス障害などにより、ロールフォワードログを保存しているディレクトリを損失した場合。
  - ◆ 意図せずにロールフォワードログ機能を無効にしてしまった場合。

- ◆ **ストリームファイルのログ機能を有効にすると、ディスクの空き容量が急速に減少します。**ストリームファイル（ログインスクリプトなど）のログ出力を有効にすると、変更があるたびに、ストリームファイル全体がロールフォワードログに複写されるためです。ストリームファイルのログ出力を無効にし、フル/インクリメンタルバックアップの際にのみストリームファイルをバックアップすると、ログファイルが大きくなるのを遅らせられます。
- ◆ **データベースの復元で最も時間を要するのは、ロールフォワードログを参照する処理です。**ロールフォワードログの容量は、ツリー構造に対して施された更新の回数に応じて増え、ストリームファイル（ログインスクリプトなど）のログ出力を有効にするとさらに増えます。  
データベースが頻繁に更新されるようであれば、バックアップの頻度を上げることも検討するとよいでしょう。こうすると、復元処理の過程でロールフォワードログを参照する処理が少なくなります。
- ◆ **ログファイル名を変更しないでください。**ログが作成されたときとファイル名が異なる場合、ログファイルは復元処理には使用できません。
- ◆ **eDirectory を削除するとロールフォワードログもすべて消えてしまいます。**いったんデータベースを削除した後、ログファイルを使って復元するのであれば、eDirectory を削除する前に、別の場所にコピーしておいてください。
- ◆ **復元が必要な場合は、復元処理の終了後にそのサーバのロールフォワードログ設定を再作成してください。**この機能を有効にし、ログの保存先を安全な場所に設定します。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

この手順が必要となるのは、復元処理の過程で、ロールフォワードログに関する設定はデフォルトに戻るためです。つまり、ロールフォワードログ機能は無効となり、保存先もデフォルトの場所になるからです。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

## ロールフォワードログの保存先

ロールフォワードログ機能を有効にした場合、その保存先を、eDirectory とは別の記憶デバイスに変更します。

保存先を設定する上で、次の点に注意してください。

- ◆ **保存先をデフォルトの場所のままにはせず、eDirectory とは別の記憶デバイス上に設定し直してください。**こうしておけば、デバイス障害のために eDirectory が失われても、復元のためにロールフォワードログにアクセスできます。

たとえば NetWare の場合、デフォルトの保存先は `sys:_netware\nds.rfl` です。ただし、ロールフォワードログ機能を有効にしたら、デフォルトの保存先を使用しないでください。eDirectory データベースが保存されている `sys:` ボリュームにログを保存しないでください。

サーバに記憶デバイスがひとつしかない場合、デバイス障害が起こればロールフォワードログも消えてしまうので、障害対策としては役に立ちません。この機能はロールフォワードログを使用しないでおく方法もあります。

ロールフォワードログの保存先を変更するには、iManager のバックアップ環境設定画面、または eMBox クライアントの `setconfig` コマンドを使用してください。ロールフォワードログはサーバ上のローカルファイルとして保存する必要があります。

- ◆ **保存先を記録してください。** ロールフォワードログの保存先を記録して、サーバのデータベースの復元が必要などに見つけられるようにしてください。これはサーバが正常で障害が発生する前に実行することが重要です。

サーバが正常に動作していれば、iManager のバックアップ環境設定画面、または eMBox クライアントの getconfig バックアップオプションで調べることができます。ただしハードウェア障害などで eDirectory が使えない状態になると、この方法でロールフォワードログの場所を調べることはできません。

サーバに障害が発生し、それを復元する場合は、eDirectory を新たにインストールすると、ロールフォワードログの保存先設定はデフォルトの場所に戻ります。このため、復元作業のために eDirectory を再インストールしたとしても、サーバの停止前にロールフォワードログをどこに保存していたか、eDirectory で調べることはできません。その場合は記録を参照して位置を調べる必要があります。

ロールフォワードログの保存先設定は、\_ndsdb.ini ファイルにも記録されています。しかしこれは eDirectory と同じディスクパーティション / ボリュームにあるため、eDirectory がある記憶デバイスに障害が起これば、ログの保存先を調べるために \_ndsdb.ini ファイルを使用することはできません。

- ◆ **ロールフォワードログの保存先へのアクセス権を制限してください。** これはセキュリティ上の問題です。見ただけで中身がわかるような形式にはなっていませんが、デコードは可能なため、重要なデータが漏洩する恐れがあります。
- ◆ **ディスクの空き容量が充分かどうか、常に監視する必要があります。** [400 ページの「ロールフォワードログのバックアップと削除」](#) を参照してください。
- ◆ **ロールフォワードログ専用のディスクパーティション / ボリュームを用意するのが最善です。** こうしておけば、ディスク容量やアクセス権を監視しやすくなります。
- ◆ **ログの保存先パスのうち、一番深い階層のディレクトリ名は eDirectory によって作成されます。** この名前は現在の eDirectory データベース名に基づいて決まります。

たとえばログの保存先を「D:\Novell\NDS\DIBFiles」と指定した場合、eDirectory データベース名が「NDS」であれば、実際の保存先ファイルは「D:\Novell\NDS\DIBFiles\nds.rfl」となります。データベースの名前を NDS から ND1 に変更した場合、ロールフォワードログのディレクトリは D:\Novell\NDS\DIBFiles\nd1.rfl\ に変更されます。

保存先の設定を変えるとその時点で新しいディレクトリができますが、ログファイルは実際にトランザクションが発生するまで作成されません。

- ◆ **復元の際は、必要なロールフォワードログをすべて同じディレクトリに集めます。** 詳細については、[401 ページの「復元処理の準備」](#) を参照してください。

## ロールフォワードログのバックアップと削除

放置しておけばロールフォワードログは次々に蓄積され、ディスクパーティション / ボリュームがいっぱいになります。ディスク容量が不足してロールフォワードログを作成できない場合は、eDirectory はそのサーバに対して応答しなくなります。定期的にログファイルをバックアップし、サーバからは削除するようにして、常に十分なディスク容量を確保するようお勧めします。

削除しても構わないロールフォワードログを判別し、バックアップを取った上で削除するには、次の手順に従います。

- 1 「最後に使用済みになった」 ロールフォワードログ名を調べてください。

最後の使用済みロールフォワードログの名前は、次の方法で調べることができます。

- ◆ iManager で [eDirectory の保守] > [バックアップ環境設定] の順にクリックし、表示されるファイル名を調べます。
- ◆ eMBox クライアントで `getconfig` バックアップコマンドを実行します。具体的な手順については [418 ページの「eMBox クライアントによるロールフォワードログの設定」](#) を参照してください。

「最後に使用済みになった」ロールフォワードログとは、トランザクション履歴の記録が終わり、今は書き出しをしていないログファイルのうち、最新のものを表します。現在は別のもっと新しいログファイルに書き出しているため、オープンしておく必要がないので、「最後に使用済みになった」ロールフォワードログと呼ばれます。一方、現在でもトランザクション履歴を書き出しているログは「使用中」で、データベースに必要なものです。

- 2 ファイルシステムのバックアップ機能を使って、ロールフォワードログをテープに保存してください。
- 3 「最後に使用済みになった」ものよりも古いロールフォワードログを削除してください。

**警告：**ロールフォワードログを削除する際にはより注意を払い、削除しようとするファイルが確実にバックアップされているか、繰り返し確認してください。

「最後に使用済みになった」とは、既にクローズされており、今では履歴の記録をしていないことを表します。サーバから削除しても構わないという意味ではありません。まだテープにバックアップしていないファイルは削除しないようにしてください。

テープに保存してあるロールフォワードログを復元のために使う場合は、次の点に注意してください。

- ◆ 復元に使用する他のロールフォワードログと同様に、ファイルシステムをバックアップしたテープから取得したログファイルは、他のログと合わせてひとつのフォルダに集めてください。このフォルダは、サーバからローカルにアクセスできる必要があります。
- ◆ テープおよびサーバに複製されたファイルのタイムスタンプを比較する必要があります。タイムスタンプに違いがある場合は、最新のサーバ上のファイルを使います。たとえば、ファイルシステムのバックアップ中に、データベースで使用されていたロールフォワードログファイルはテープに完全に保存されません。最新の完全なファイルはサーバに格納されています。

## 注意：eDirectory を削除するとロールフォワードログも削除される問題

サーバから eDirectory を削除すると、ロールフォワードログのディレクトリおよびその中身もすべて削除されます。いったんデータベースを削除した後、ログファイルを使って復元するのであれば、eDirectory を削除する前に、別の場所にコピーしておいてください。

## 復元処理の準備

eDirectory データベースの復元作業で最も大切なのは、復元が完全に行われたかどうか確認することです。作業に先立ち、[402 ページの「復元作業の前提条件」](#)の説明に従って、必要な準備を行います。必要なバックアップファイルを揃える手順を確認するには、[403 ページの「復元に必要なバックアップファイルの収集」](#)を参照してください。

## 復元作業の前提条件

- 復元するサーバとレプリカを共有しているサーバはすべて、稼動状態で、通信できるようにしておかなければなりません。これは、復元後の検証処理で、同じレプリカリングに属するサーバ間の整合性を確認するために必要です。
- 必要な次のバックアップファイルをすべて収集してください。
  - ◆ フルバックアップおよびそれ以降のインクリメンタルバックアップのファイルを、復元するサーバの、1つのディレクトリ内に集めます。
  - ◆ 最後にバックアップを取って以降のロールフォワードログをすべて、同じサーバ上のもう1つのディレクトリにまとめておきます。

このサーバがレプリカリングに属している場合、最後にバックアップを取った時点以降のロールフォワードログをすべて、ひとつのディレクトリ内にまとめておきます。ファイル名はログ生成時と同じにしておかなければなりません。

403 ページの「復元に必要なバックアップファイルの収集」を参照してください。

**注：**バックアップファイルがない場合は、XBrowse を使って、サーバ情報の復元に必要な情報を eDirectory に問い合わせてください。この作業は、サーバオブジェクトやその関連オブジェクトをツリーから削除する前に実行する必要があります。Xbrowse の詳細については、[Novell Support Web site, Solution 2960653 \(http://support.novell.com/servlet/tidfinder/2960653\)](http://support.novell.com/servlet/tidfinder/2960653) を参照してください。

- eDirectory を再インストールし、仮のツリーで稼動させておきます。

最初はサーバを仮のツリーで稼動させます。最終的には障害が起きる前と同じ名前のサーバに復元しようとしているわけですが、まだ復元作業が完了していない段階で本来のツリーに組み込んでしまうと、混乱が生じてしまうからです。データベースの復元処理が完了してから、本来のツリーにサーバを組み入れることとなります。
- (状況によって実行) このサーバでロールフォワードログ機能を使うためには、復元後に改めて有効に切り替え、障害対策のための書き出し先も設定し直して、ロールフォワードログの環境設定を再作成する必要があります。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

復元処理の過程で、ロールフォワードログの機能は無効になり、ログ保存先の設定もデフォルトに戻ってしまいます。

フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。
- (状況によって実行) IPアドレスを指定してこのサーバにアクセスするアプリケーションやオブジェクトがある場合は、元と同じ IP アドレスを設定してください。
- (NetWare の場合のみ) 復元されたサーバの名前を、障害前と同じにしてください。同じ名前を使用しないと、復元後に「ボリュームオブジェクトが不正」などのエラーが起こる可能性があります。

復元する NetWare サーバ名を変更するには、`autoexec.ncf` ファイルを書き替え、サーバを再起動してください。
- (NetWareのみ) ファイルシステムデータおよび eDirectory を復元する際は、ファイルシステム権利を元どおりに戻せるよう、正しい手順で作業してください。eDirectory を先に復元し、その後でファイルシステムを復元してください。この手順の詳細については、[396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」](#)を参照してください。

復元の過程で、eDirectory Backup eMTool はまず、フルバックアップファイルからの復元を試みます。それが済むと、Backup eMTool でインクリメンタルバックアップファイルの名前を入力するよう求められます。その際、次に適用するべきファイルの ID が提示

されます。インクリメンタルバックアップファイルからの復元が終わると、今度はロールフォワードログを参照しての復元処理が始まります (388 ページの「Backup eMTool による復元作業の概要」も参照してください)。

必要なファイルをすべて揃えた後、iManager や eMBox クライアントから復元処理を起動します。420 ページの「eMBox クライアントによるバックアップファイルの復元作業」または 409 ページの「iManager によるバックアップファイルの復元作業」を参照してください。

## 復元に必要なバックアップファイルの収集

- 1 ファイルシステムをバックアップしたテープから、eDirectory フルバックアップファイルを、サーバ上の適切なディレクトリにコピーしてください。

最後に取ったフルバックアップの ID は、Backup eMTool のログファイルで確認できます。

- 2 同様に、一連のインクリメンタルバックアップファイルを、サーバ上の適切なディレクトリにコピーしてください。

必要なインクリメンタルバックアップファイルは、フルバックアップファイルのヘッダ部で確認できます。「next\_inc\_file\_ID」属性として、次のインクリメンタルバックアップファイルの ID が記述されています。この値は、インクリメンタルバックアップファイルのヘッダ部にある「incremental\_file\_number」属性に対応します。ヘッダの形式について詳しくは、389 ページの「バックアップファイルのヘッダ書式」を参照してください。

**警告:** バックアップファイルを開く場合でもヘッダを確認するだけにとどめ、保存や変更はしないようにしてください。ファイルの一部が切り捨てられてしまうことがあります。ほとんどのアプリケーションではバイナリデータを正確に保存することはできません。

インクリメンタルバックアップファイルにはそれぞれ、次のインクリメンタルバックアップファイルの ID が記載されています。

この ID も Backup eMTool のログファイルで確認できます。

同じ名前のファイルがいくつもあって、ひとつのディレクトリにまとめるためにファイル名を変更しているような場合、ID はその識別に不可欠です。たとえば無人でのバックアップにいつも同じバッチファイルを使っていて、バックアップファイル名が常に同じであるような場合です。ヘッダ部の ID を見れば、ファイル名が変わっていても適切なファイルを判別できます。

- 3 (状況によって実行)ロールフォワードログ機能を有効にしていた場合は、最後のバックアップ以降のロールフォワードログを、生成時のファイル名のまま、サーバ上の適切なディレクトリに集めてください。

このサーバがレプリカリングに属している場合は、ロールフォワードログを使った復元処理が必須です。ロールフォワードログがすべて揃っていない場合、復元後の検証処理で失敗してしまいます。リング内の他のレプリカと、遷移ベクトルが一致しないからです。デフォルトでは、復元した eDirectory データベースが他のレプリカと整合が取れていない場合、そのままではオープンできません。

テキストエディタで最新のバックアップファイルを開き、ヘッダの「current\_log」属性を読んで、最初に必要なロールフォワードログを特定します。この作業を繰り返して、続くすべてのロールフォワードログを集めます。

**警告:** バックアップファイルを開く場合でもヘッダを確認するだけにとどめ、保存や変更はしないようにしてください。ファイルの一部が切り捨てられてしまうことがあります。ほとんどのアプリケーションではバイナリデータを正確に保存することはできません。

必要なロールフォワードログがすべて 1ヶ所にまとまっているとは限りません。よく確認して、すべて同じディレクトリに揃えてください。ロールフォワードログは、次の理由から複数の場所に格納されている場合があります。

- ◆ 最後にeDirectoryのフル/インクリメンタルバックアップを実行してから、ロールフォワードログの保存先を変更した場合。
- ◆ ファイルシステムのバックアップを使用して、ロールフォワードログをテープにバックアップした後で、空きディスク容量を確保するためにそれらのファイルを削除した場合。

テープにバックアップされたロールフォワードログを取得する場合は、データが最新のセットであることを確認してください。テープおよびサーバに複製されたファイルのタイムスタンプを比較する必要があります。ファイルシステムのバックアップ中に、データベースで使用されていたロールフォワードログファイルはテープに完全に保存されません。最新の完全なファイルはサーバに格納されています。

- ◆ 最後にバックアップを実行してから eDirectory データベースの名前を変更した (NDS から ND1 に変更した場合など)。この変更により、ロールフォワードログのパスの最後のディレクトリ名が変更されます。

たとえばログの保存先を「D:\novell\nds\dibfiles\」と指定した場合、eDirectory データベース名が「NDS」であれば、実際の保存先ファイルはディレクトリ「D:\novell\nds\dibfiles\nds.rfl\」となります。データベースの名前を NDS から ND1 に変更した場合、ロールフォワードログのディレクトリは D:\Novell\nds\dibfiles\nd1.rfl\ に変更されます。

**重要:** 必要なロールフォワードログがすべてそろっていることを確認してください。Backup eMTool では、ロールフォワードログがすべてそろっているかどうか確認できません。ロールフォワードログは順番に開かれて使用されます。指定したディレクトリ内に次のロールフォワードログが見つからない場合は、復元プロセスが中止されます。必要なロールフォワードログがすべてそろっていなければ復元は完了しません。

## Novell iManager を使ったバックアップ / 復元作業

バックアップやその環境設定、復元の作業は、Novell iManager から eDirectory Backup eMTool を呼び出す形で実行できます。iManager を使うことにより、ファイアウォールの外側からでも、Web ブラウザ画面で操作できます。Novell iManager の詳細については、『*Novell iManager 2.0.x Administration Guide(Novell iManager 2.0.x 管理ガイド)* (<http://www.novell.com/documentation/japanese/imanager20/index.html>)』を参照してください。

ただし、コールドバックアップ (データベースをいったん停止したフルバックアップ)、無人バックアップ、高度な復元機能は、iManager からは実行できません。これらの作業は eMBox クライアントで実行することになります。詳しくは [412 ページの「eMBox クライアントを使ったバックアップ / 復元作業」](#) を参照してください。

eDirectory のバックアップ / 復元作業に先立ち、[382 ページの「eDirectory のバックアップ処理に関する確認事項」](#) を参照して問題点を確認し、効率的に作業できるようにしてください。

このセクションでは、次のトピックについて説明します。

- ◆ [405 ページの「iManager による手動バックアップ」](#)
- ◆ [407 ページの「iManager によるロールフォワードログの設定」](#)
- ◆ [409 ページの「iManager によるバックアップファイルの復元作業」](#)



## iManager による手動バックアップ

iManager のブラウザ画面から [バックアップ] を使用して、eDirectory データベースをサーバにバックアップします。フルバックアップ、インクリメンタルバックアップのどちらも実行可能です。

バックアップファイルには、eDirectory をその時点の状態に復元するために必要な情報がすべて含まれています。また、処理結果は所定のログファイルに記録されます。

iManager から実行できるのは「ホット」バックアップです。つまり、バックアップ処理中も eDirectory データベースは開いたままで、通常どおり利用しながら、バックアップ開始時点の状態を完全に保存できます。

なお、コールドバックアップ ( データベースを停止してのバックアップ ) や無人バックアップを実行するには、eMBox クライアントを使用する必要があります。412 ページの「eMBox クライアントによる手動バックアップ」および 415 ページの「バッチファイルと eMBox クライアントによる無人バックアップ」を参照してください。

eDirectory のバックアップ / 復元作業に先立ち、382 ページの「eDirectory のバックアップ処理に関する確認事項」を参照して問題点を確認し、効率的に作業できるようにしてください。

### 前提条件

- ❑ eDirectory 以外にも追加でバックアップしたいファイルがあれば、それを列挙したインクルードファイルを作っておいてください。

iManager の設定画面で該当するチェックボックスをオンにすれば、NICI ファイルやストリームファイルもバックアップできます。NICI ファイルは常にバックアップするようお勧めします。

それ以外にたとえば autoexec.ncf などをバックアップしたい場合は、そのパスとファイル名をインクルードファイルに列挙します。複数のファイルがある場合はセミコロンで区切ります。改行 ( ハードリターン ) や空白を含めないようにしてください ( 例 : 「 sys:\system\autoexec.ncf;sys:\etc\hosts;」 ) 。

- ❑ eDirectory のバックアップ後すぐに、ファイルシステムのバックアップ作業を行い、テープに保存できるよう準備してください (Backup eMTool による処理では、サーバ上にバックアップファイルができるだけです)。

ヒント : コピー先記憶デバイスに容量の制約がある場合は、あらかじめ eDirectory バックアップファイルの最大サイズを設定すると便利です。また、バックアップファイルの作成後、サードパーティ製ファイル圧縮ツールを使う方法もあります。80% 程度は圧縮できます。

- ❑ ロールフォワードログを作成するのであれば、バックアップを行う前にこの機能を有効にしてください。

レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。

ロールフォワードログの詳細については、397 ページの「ロールフォワードログを使用する」を参照してください。また、この機能を有効にする手順については、407 ページの「iManager によるロールフォワードログの設定」を参照してください。


- ❑ 複数サーバ環境のツリーの場合、このサーバとレプリカを共有するサーバすべてについて、eDirectory 8.5 以降にアップグレードする必要があります。

詳細については、395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」を参照してください。

## 操作手順

iManager を使って eDirectory データベースをバックアップする手順を次に示します。

ヒント : iManager で使用できるオプションについてはオンラインヘルプを参照してください。

- 1 [役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [バックアップ] の順にクリックします。
- 3 バックアップの対象サーバを指定し、[次へ] をクリックします。
- 4 バックアップを実行するサーバのユーザ名、パスワード、コンテキストを指定し、[次へ] をクリックします。
- 5 バックアップファイルのオプションを指定し、[次へ] をクリックします。

最後にバックアップを実行した後の差分のみをバックアップしたい場合は、[インクリメンタルバックアップを実行します] をクリックしてください。

画面例を次に示します。

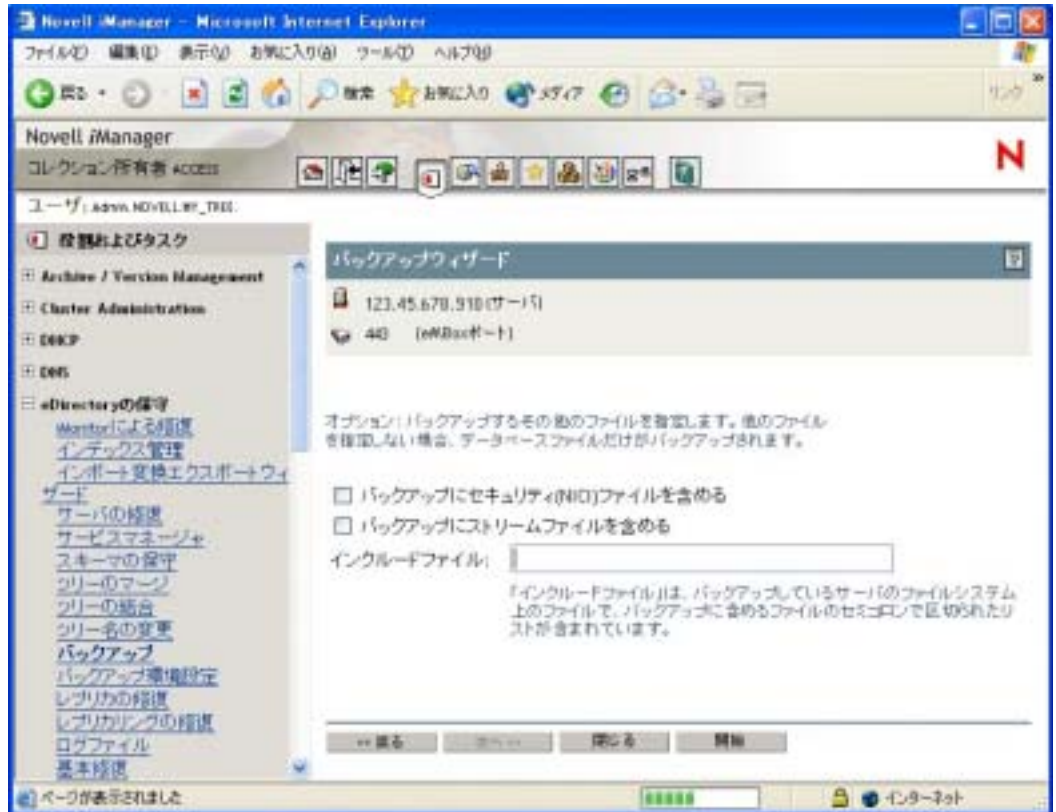


- 6 追加でバックアップしたいファイルがあればここで指定します。

追加するファイルが指定されていない場合、eDirectory データベースのみがバックアップされます。

NICI ファイルは常にバックアップするようお勧めします。

画面例を次に示します。



7 表示される指示に従って、バックアップを完了します。

8 eDirectory のバックアップ処理が終了したら、すぐにファイルシステムのバックアップ作業を行い、テープに保存します (Backup eMTool による処理では、サーバ上にバックアップファイルができるだけです)。


## iManager によるロールフォワードログの設定

ブラウザから [バックアップ環境設定] を使用して、ロールフォワードログに関する設定を変更します。次の設定ができます。

- ◆ ロールフォワードログ機能の有効/無効の切り替え  
レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないことになります。
- ◆ ロールフォワードログの保存先ディレクトリの変更
- ◆ ロールフォワードログのファイルサイズの最小値、最大値の設定
- ◆ 現在使用中のログ、既書き出しを終えた最新のログの判別
- ◆ ストリームファイルをロールフォワードログに含めるかどうかの切り替え

ロールフォワードログの詳細については、[397 ページ](#)の「[ロールフォワードログを使用する](#)」を参照してください。

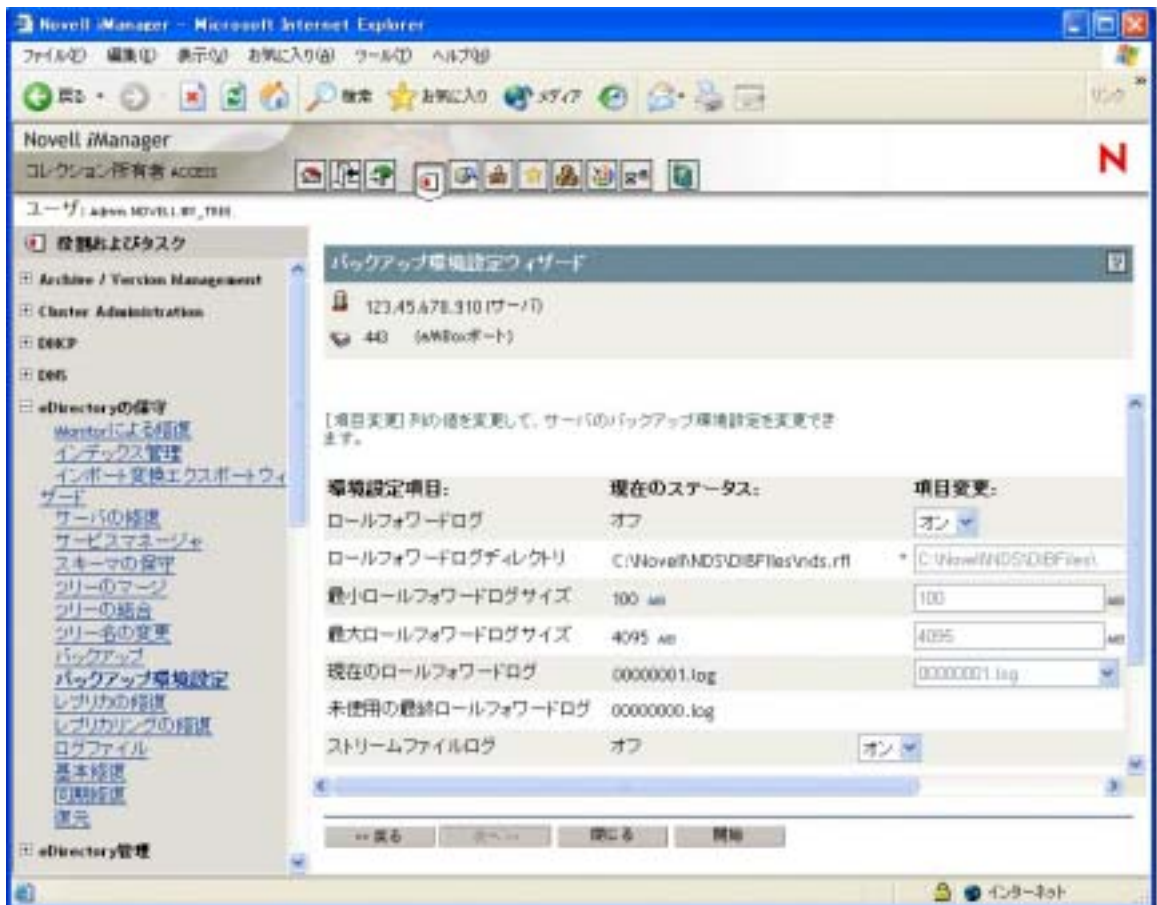
ヒント : iManager で使用できるオプションについてはオンラインヘルプを参照してください。

- 1 [役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [バックアップ環境設定] の順にクリックします。
- 3 設定を変更するサーバを指定し、[次へ] をクリックします。
- 4 設定を変更するサーバのユーザ名、パスワード、コンテキストを指定し、[次へ] をクリックします。
- 5 必要に応じてサーバのバックアップ環境設定を変更します。

**警告:** ロールフォワードログ機能を有効にしたら、デフォルトの保存先を使用しないでください。障害対策のためには、eDirectory とは別のディスクパーティション / ボリューム、別の記憶デバイスを指定してください。ロールフォワードログディレクトリは、バックアップ環境設定を変更するサーバ上である必要があります。

**重要:** ロールフォワードログ機能を有効にする場合、ログを保存するボリュームのディスク容量を常に監視してください。これを怠ると、ログの容量は増える一方なので、ディスクパーティション / ボリュームがあふれてしまう恐れがあります。ディスク容量が不足してロールフォワードログを作成できない場合は、eDirectory はそのサーバに対して応答しなくなります。書き出しが終わったロールフォワードログは、定期的にバックアップし、サーバから削除するようお勧めします。400 ページの「**ロールフォワードログのバックアップと削除**」を参照してください。

画面例を次に示します。



- 6 表示される指示に従って、操作を完了します。

## iManager によるバックアップファイルの復元作業

ブラウザから [復元] を使用して、保存されたバックアップファイルのデータから eDirectory データベースを復元します。処理結果は所定のログファイルに記録されます。

復元処理の詳細については、[388 ページの「Backup eMTool による復元作業の概要」](#)を参照してください。

高度な復元機能は eMBox クライアントから実行する必要があります。詳しくは [412 ページの「eMBox クライアントを使ったバックアップ / 復元作業」](#)を参照してください。


### 前提条件

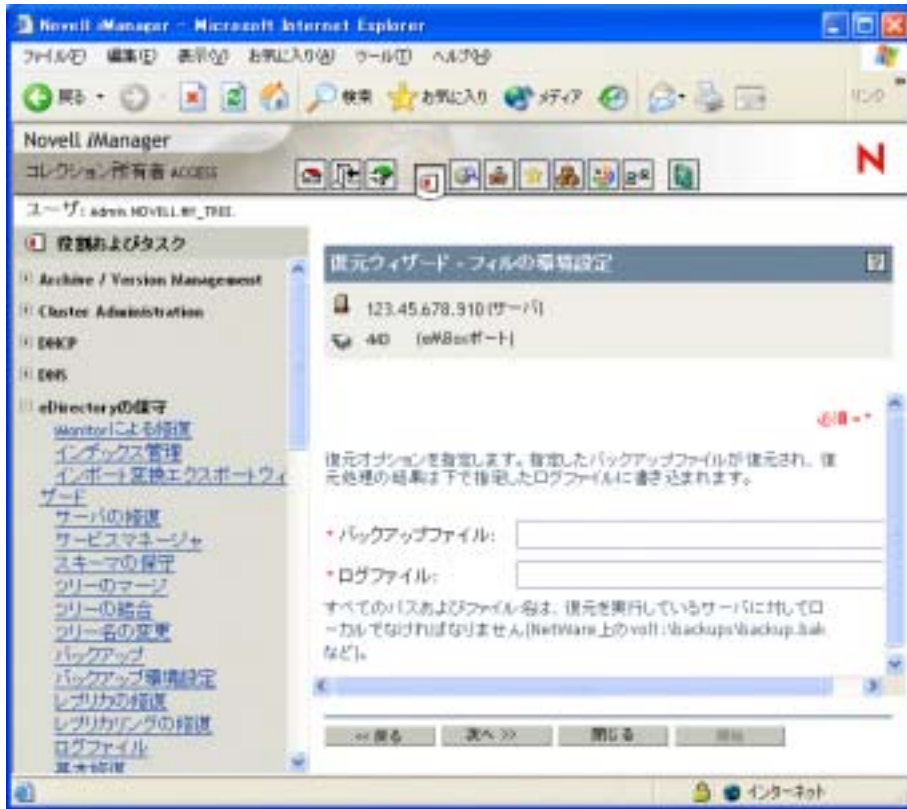
- ❑ 必要なバックアップファイルをすべて、復元対象サーバ上の、適当なディレクトリに集めておく必要があります。  
[401 ページの「復元処理の準備」](#) および [403 ページの「復元に必要なバックアップファイルの収集」](#) を参照してください。
- ❑ eDirectory を復元対象のサーバにインストールし、稼働させておいてください。  
たとえば記憶デバイスの障害の場合、デバイスを交換し、改めて eDirectory をインストールすることになります。故障したサーバごとと交換する、あるいは単に新しいサーバに移行する場合は、新しいサーバにオペレーティングシステムをインストールした上で、eDirectory も準備します。
- ❑ 復元処理の詳細については、[388 ページの「Backup eMTool による復元作業の概要」](#)を参照してください。
- ❑ (NetWareのみ)ファイルシステムデータおよびeDirectoryを復元する際は、ファイルシステム権利を元どおりに戻せるよう、正しい手順で作業してください。eDirectory を先に復元し、その後でファイルシステムを復元してください。この手順の詳細については、[396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」](#)を参照してください。

### 操作手順

ヒント : iManager で使用できるオプションについてはオンラインヘルプを参照してください。

iManager を使って eDirectory データベースを復元する手順を次に示します。

- 1 必要なバックアップファイルを集めておきます。詳しくは [401 ページの「復元処理の準備」](#) を参照してください。
- 2 [役割およびタスク] ボタン  をクリックします。
- 3 [eDirectory の保守] > [復元] の順にクリックします。
- 4 復元対象サーバを指定し、[次へ] をクリックします。
- 5 復元を実行するサーバのユーザ名、パスワード、コンテキストを指定し、[次へ] をクリックします。
- 6 バックアップファイル名、ログファイル名を指定し、[次へ] をクリックします。  
画面例を次に示します。



7 必要な復元オプションを指定し、[次へ] をクリックします。

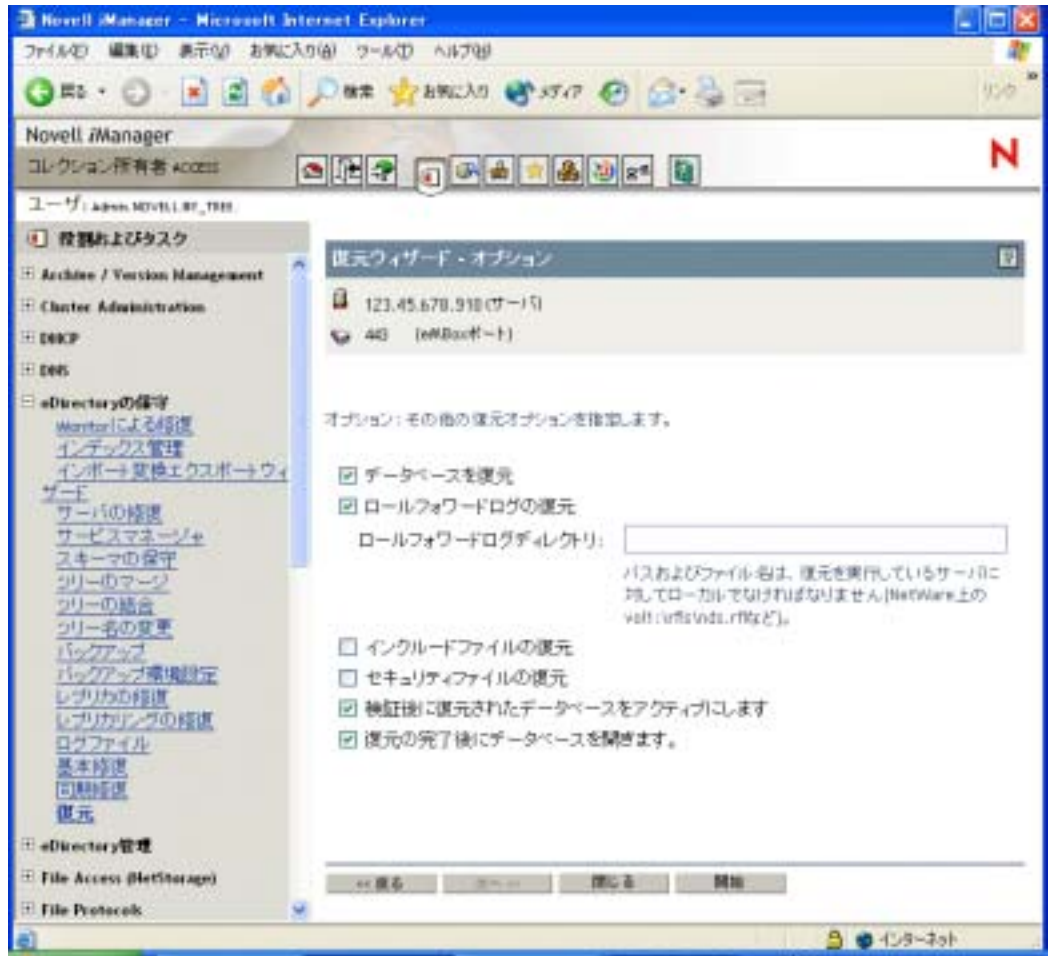
通常、少なくとも次のチェックボックスはオンにする必要があります。

- ◆ データベースを復元
- ◆ 検証後に復元されたデータベースをアクティブにします
- ◆ 復元の完了後にデータベースを開きます
- ◆ セキュリティファイルの復元 (NICI ファイルの復元)

NICI ファイルは必ずバックアップしておくようお勧めします。これがないと、復元に成功しても、暗号化されたファイルは読めません。

ロールフォワードログを使って復元する場合は、そのフルパスを指定しなければなりません。これには、eDirectory が自動的に追加するディレクトリ名 (通常は「nds.rfl」) も含まれます。詳しくは 399 ページの「ロールフォワードログの保存先」を参照してください。

画面例を次に示します。



8 表示される指示に従って、復元を完了します。

復元後の検証に失敗した場合の対処については、[432 ページの「復元後の検証処理に失敗した場合の対処方法」](#)を参照してください。

注：レプリカリング中に eDirectory 8.5 より前のバージョンが稼動しているサーバがある場合、復元処理は失敗します。エラーコードは -666、すなわち「DS バージョンの不整合」となります。この場合の対処方法については、[395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#)を参照してください。

9 NCSIセキュリティファイルを復元した場合は、復元完了後に NCSI を再初期化するため、サーバを再起動します。

10 ここでサーバが通常どおり要求に応答することを確認しておきます。

11 (状況によって実行) このサーバでロールフォワードログ機能を使うためには、改めて有効に切り替え、障害対策のための書き出し先も設定し直して、ロールフォワードログの環境設定を再作成する必要があります。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

この手順が必要となるのは、復元処理の過程で、ロールフォワードログに関する設定はデフォルトに戻るためです。つまり、ロールフォワードログ機能は無効となり、保存先もデフォルトの場所になるからです。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。

以上で復元作業が終了しました。NICI の再初期化も済んでいるので、暗号化された情報にもアクセスできます。ロールフォワードログ機能を使用する場合は、今後の障害に備えるため、再びこの機能を有効にし、フルバックアップを取っておいてください。

## eMBox クライアントを使ったバックアップ / 復元作業

eMBox クライアントはコマンドライン Java クライアントで、これを使用すると eDirectory Backup eMTool などの eMBox ツールにアクセスできます。複数サーバ環境でも、ファイアウォール越しのアクセスができれば、1 台のコンピュータからバックアップ、復元、ロールフォワードログの設定ができます。

eMBox クライアントはバッチモードで実行できるため、eDirectory Backup eMTool を使用して無人バックアップを行うことができます。

eMBoxClient.jar ファイルは、eDirectory の一部としてサーバにインストールされます。それ以外にも、Sun JVM 1.3.1 が動作する環境であれば、eMBoxClient.jar をコピーして動かすことができます。詳細については、[479 ページの「eDirectory Management Toolbox」](#) および [481 ページの「ワークステーションで eMBox クライアントを実行する」](#) を参照してください。

eDirectory のバックアップ / 復元作業に先立ち、[382 ページの「eDirectory のバックアップ処理に関する確認事項」](#) を参照して問題点を確認し、効率的に作業できるようにしてください。

このセクションでは、次のトピックについて説明します。

- ◆ [412 ページの「eMBox クライアントによる手動バックアップ」](#)
- ◆ [415 ページの「バッチファイルと eMBox クライアントによる無人バックアップ」](#)
- ◆ [418 ページの「eMBox クライアントによるロールフォワードログの設定」](#)
- ◆ [420 ページの「eMBox クライアントによるバックアップファイルの復元作業」](#)
- ◆ [423 ページの「バックアップ / 復元のコマンドラインオプション」](#)

## eMBox クライアントによる手動バックアップ

eMBox クライアントを使って、eDirectory データベースの中身を、指定したファイルにバックアップすることができます。バックアップファイルには、eDirectory をその時点の状態に復元するために必要な情報がすべて含まれています。また、処理結果は所定のログファイルに記録されます。

eDirectory のバックアップ / 復元作業に先立ち、[382 ページの「eDirectory のバックアップ処理に関する確認事項」](#) を参照して問題点を確認し、効率的に作業できるようにしてください。

eMBox クライアントを使うと次のような作業ができます。

- ◆ データベースを開いたままで、フル / インクリメンタルバックアップ (ホットバックアップ)。  
「ホット」バックアップの場合、バックアップ処理中も eDirectory データベースは開いたままで、通常どおり利用しながら、バックアップ開始時点の状態を完全に保存できます。
- ◆ コールドバックアップ (データベースをいったん停止してフルバックアップ)。  
この機能は、ハードウェアをアップグレードする、あるいは新規サーバ (同じオペレーティングシステムが動作するもの) に移行する場合に有用です。詳しくは [460 ページの「ハードウェアのアップグレードやサーバの交換」](#) を参照してください。



- ◆ バックアップ後、データベースを閉じたままにしてロックする設定。
- ◆ バックアップファイルの最大サイズの設定。

これらの作業を無人で実行するための手順については、[415 ページの「バッチファイルと eMBox クライアントによる無人バックアップ」](#)を参照してください。

## 前提条件

- バックアップ処理を起動するコンピュータに、ファイル `eMBoxClient.jar` があることを確認してください。

このファイルは、eDirectory の一部としてサーバにインストールされます。それ以外にも、Sun JVM 1.3.1 が動作する環境であれば、`eMBoxClient.jar` をコピーして動かすことができます。複数サーバ環境でも、ファイアウォール越しのアクセスが可能であれば 1 台のコンピュータから作業できます。詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

- ロールフォワードログを作成するのであれば、バックアップを行う前にこの機能を有効にしてください。

レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。また、この機能を有効にする手順については、[418 ページの「eMBox クライアントによるロールフォワードログの設定」](#)を参照してください。

- eDirectory 以外にも追加でバックアップしたいファイルがあれば、それを列挙したインクルードファイルを作っておいてください。

NICI ファイルやストリームファイルもスイッチを使用してバックアップできます。NICI ファイルは常にバックアップするようお勧めします。

それ以外にたとえば `autoexec.ncf` などをバックアップしたい場合は、そのパスとファイル名をインクルードファイルに列挙します。複数のファイルがある場合はセミコロンで区切ります。改行 (ハードリターン) や空白を含めないようにしてください (例: 「`sys:\system\autoexec.ncf;sys:\etc\hosts;`」)。

- eDirectory のバックアップ後すぐに、ファイルシステムのバックアップ作業を行い、テープに保存できるよう準備してください (Backup eMTool による処理では、サーバ上にバックアップファイルができるだけです)。

**ヒント:** コピー先記憶デバイスに容量の制約がある場合は、あらかじめ eDirectory バックアップファイルの最大サイズを設定すると便利です。その場合、バックアップコマンドの「-s」オプションを使い、バイト単位で指定します。また、バックアップファイルの作成後、サードパーティ製ファイル圧縮ツールを使う方法もあります。80% 程度は圧縮できます。

- コマンドラインオプションについては、[423 ページの「バックアップ / 復元のコマンドラインオプション」](#)を参照してください。

- 複数サーバ環境のツリーの場合、このサーバとレプリカを共有するサーバすべてについて、eDirectory 8.5 以降にアップグレードする必要があります。

詳細については、[395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#)を参照してください。

## 操作手順

eMBox クライアントを使って eDirectory データベースをバックアップする手順を次に示します。

### 1 eMBox クライアントを対話式モードで起動します。

- ◆ NetWare、UNIX の場合：コマンドラインから「**edirutil -i**」と入力します。
- ◆ Windows の場合：「**drive\novell\nds\embox\edirutil.exe -i**」を実行します。

edirutil ファイルは、eMBox クライアントを実行するためのショートカットです。Java の実行形式ファイルと、eMBox クライアントのインストール先ディレクトリがパラメータとして記述されているほか、NetWare の場合は「-ns」オプションもついています。(Java 実行ファイルの場所は、[481 ページの「eMBox クライアント用にパスおよびクラスパスを設定する」](#)で示しているように手動で入力することもできます)。

正常に起動されると、「eMBox Client>」というプロンプトが現れます。

### 2 バックアップの対象サーバにログインします。次のように入力してください。

```
login -s server_name_or_IP_address -p port_number  
-u username.context -w password
```

たとえば Windows の場合、次のようになります。

```
login -s 151.155.111.1 -p 8009 -u admin.mycompany -w mypassword
```

セキュア接続が確立できないというエラーが表示される場合は、[487 ページの「eMBox クライアントを使用してセキュア接続を確立する」](#)に表示されている JSSE ファイルがシステム上にない可能性があります。

指定するポート番号が分からない場合は、[488 ページの「eDirectory ポート番号を確認する」](#)を参照してください。

eMBox クライアントはログインが成功したかどうかを表示します。

### 3 eMBox クライアントのプロンプトが出たら、次の形式でバックアップコマンドを入力します。

```
backup -b -f backup_filename_and_path -l backup_log_filename_and_path -u  
include_file_filename_and_path -e -t -w
```

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。「-e」オプションを指定して、NICI ファイルもバックアップするようお勧めします。暗号化されたファイルを復元し、中身を読むために必要です。

たとえば Windows の場合、次のようになります。

```
backup -b -f c:\backups\8_20_2001.bak -l c:\backups\backup.log -u  
c:\backups\myincludefile.txt -e -t -w
```

この例では、フルバックアップを取ることに (-b)、バックアップファイルを c:\backups\8\_20\_2001.bak とすること、処理結果を c:\backups\backup.log に出力すること、さらに、次のデータベース以外のファイルもバックアップすることを指定しています。

- ◆ 管理者があらかじめ作成したインクルードファイル (c:\backups\myincludefile.txt) に列挙されたバックアップ対象のファイル (「-u」オプションで指定)。
- ◆ NICI ファイル (「-e」オプションで指定)。
- ◆ ストリームファイル (「-t」オプションで指定)。

さらにこの例では「-w」オプションが指定されているため、同じ名前のバックアップファイルがあれば上書きされます。

eMBox クライアントはバックアップが成功したかどうかを表示します。

- 4 サーバからログアウトするには、次のコマンドを入力します。

```
logout
```

- 5 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

- 6 eDirectory のバックアップ処理が終了したら、すぐにファイルシステムのバックアップ作業を行い、テープに保存します (Backup eMTool による処理では、サーバ上にバックアップファイルができるだけです)。

## バッチファイルと eMBox クライアントによる無人バックアップ

バッチファイルを使用して、eMBox クライアントによる eDirectory の無人バックアップを実行します。たとえば週 1 回フルバックアップ、毎晩インクリメンタルバックアップを取る、といった運用が可能です。

バッチモードで eMBox クライアントを実行するには、システムバッチファイルを使う、eMBox クライアントの内蔵バッチファイルを使う、両者を組み合わせて使う、という方法があります。詳細については、[484 ページの「eMBox コマンドラインクライアントをバッチモードで実行する」](#)を参照してください。

ここではシステムバッチファイルを使う方法を解説します。

### 前提条件

- バッチファイルを自動で実行する方法については、ご使用のオペレーティングシステムのマニュアルまたはサードパーティ製スケジューリングソフトウェアのマニュアルを参照してください。

**注：**NetWare の場合、サードパーティ製ソフトウェアのほか、Novell Support Web サイトで提供している [cron.nlm](http://support.novell.com/servlet/tidfinder/2939440) (<http://support.novell.com/servlet/tidfinder/2939440>) も使えます。

- バックアップ処理を起動するコンピュータに、ファイル eMBoxClient.jar があることを確認してください。

このファイルは、eDirectory の一部としてサーバにインストールされます。それ以外にも、Sun JVM 1.3.1 が動作する環境であれば、eMBoxClient.jar をコピーして動かすことができます。複数サーバ環境でも、ファイアウォール越しのアクセスが可能であれば 1 台のコンピュータから作業できます。詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

- ロールフォワードログを作成するのであれば、バックアップを行う前にこの機能を有効にしてください。

レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。また、この機能を有効にする手順については、[418 ページの「eMBox クライアントによるロールフォワードログの設定」](#)を参照してください。

- eDirectory 以外にも追加でバックアップしたいファイルがあれば、それを列挙したインクルードファイルを作っておいてください。

NICI ファイルやストリームファイルもスイッチを使用してバックアップできます。NICI ファイルは常にバックアップするようお勧めします。

それ以外にたとえばautoexec.ncfなどをバックアップしたい場合は、そのパスとファイル名をインクルードファイルに列挙します。複数のファイルがある場合はセミコロンで区切ります。改行(ハードリターン)や空白を含めないようにしてください(例:「sys:\system\autoexec.ncf;sys:\etc\hosts;」)。

- eDirectory のバックアップ後すぐに、ファイルシステムのバックアップ作業を行い、テープに保存できるよう準備してください(Backup eMTool による処理では、サーバ上にバックアップファイルができるだけです)。

**ヒント:** コピー先記憶デバイスに容量の制約がある場合は、あらかじめ eDirectory バックアップファイルの最大サイズを設定すると便利です。また、バックアップファイルの作成後、サードパーティ製ファイル圧縮ツールを使う方法もあります。80% 程度は圧縮できます。

- コマンドラインオプションについては、[423 ページの「バックアップ / 復元のコマンドラインオプション」](#)を参照してください。

## 操作手順

- 1 サーバをバックアップするためのシステムバッチファイルを作成します。次の書式で、1 行に 1 サーバ分のコマンドを記述してください。

Windows、UNIX の場合、通常は次の書式を使用します。

```
java -cp path/eMBoxClient.jar embox -s server_name -p port_number -u  
username.context -w password -t backup.backup -b -f  
backup_filename_and_path -l backup_log_filename_and_path -u  
include_file_filename_and_path -e -t -w
```

NetWare でも同様ですが、オプション「-nsac」を追加してください。他のプラットフォームでは、このオプションを指定してはいけません。

```
java -nsac -cp path/eMBoxClient.jar embox -s server_name -p port_number -  
u username.context -w password -t backup.backup -b -f  
backup_filename_and_path -l backup_log_filename_and_path -u  
include_file_filename_and_path -e -t -w
```

具体例とその解説については、[417 ページの「無人バックアップ用システムバッチファイルの例」](#)を参照してください。

毎晩実行するインクリメンタルバックアップにも同様のバッチファイルが使えますが、オプション「-b」を削除し、代わりに「-i」を追加してください。フルバックアップとインクリメンタルバックアップで、保存先バックアップファイル名を異なるものにしておく方がよいでしょう。

指定するポート番号が分からない場合は、[488 ページの「eDirectory ポート番号を確認する」](#)を参照してください。セキュア接続を使用する場合は、[487 ページの「eMBox クライアントを使用してセキュア接続を確立する」](#)を参照してください。eMBox クライアントの内蔵バッチファイルの使い方については、[484 ページの「eMBox コマンドラインクライアントをバッチモードで実行する」](#)を参照してください。

- 2 このバッチファイルを定期的に起動するよう設定します。具体的な設定方法については、オペレーティングシステムまたはサードパーティ製ソフトウェアの資料を参照してください。

- 3 eDirectory のバックアップ後すぐに、ファイルシステムのバックアップ作業を行い、テープに保存できるよう準備してください  
Backup eMToolによる処理では、サーバ上にバックアップファイルができるだけです。
- 4 バックアップが正常に実行されているか、ログファイルで定期的に確認してください。

### 無人バックアップ用システムバッチファイルの例

次の2つの例を紹介します。

- ◆ [417 ページの「NetWare 用のバッチファイル例」](#)
- ◆ [418 ページの「Windows 用のバッチファイル例」](#)

#### NetWare 用のバッチファイル例

```
java -nsac -cp sys:\system\embox\emBoxClient.jar embox -s 10.10.1.200 -p 8008  
-u admin.mycontainer -w mypassword -n -t backup.backup -b -f  
sys:\system\backup\backup.bak -l sys:\system\backup\backup.log -u  
sys:\system\backup\includefile.txt -e -t -w
```

この例には次のオプションが指定されています。

- ◆ NetWare では、コマンド名「java」のすぐ後に「-nsac」を指定します（それ以外のプラットフォームでは指定しないでください）。

**警告:** NetWare サーバでは「-ns」も指定しなければ異常終了します。

「-ns」オプションを使用すると、新規画面が開きます。

これに続く「ac」オプションは、バッチファイルの処理が終了したらこの画面を閉じるよう指定するものです。NetWare のバッチファイルでこれを指定しないと、バックアップ処理のたびにサーバ上に画面が開き、そのまま残ってしまいます。

- ◆ フルバックアップを取る指定 (-b)。
- ◆ インクルードファイルの指定 (-u)。これは必要な場合のみ指定してください。データベース以外にもファイルをバックアップしたい場合に使います。インクルードファイルはあらかじめ用意しておいてください。
- ◆ NICI セキュリティファイル (-e)、ストリームファイル (-t) もバックアップする指定。NICI ファイルは常にバックアップするようお勧めします。
- ◆ 同じ名前のバックアップファイルがあれば上書きする指定 (-w)。

**重要:** 同じバッチファイルを繰り返し実行するなど、同じ名前のバックアップファイルがある場合、「-w」を指定しないと正常にバックアップされません。

バッチモードでは、「-w」の指定がなければ、同じ名前のファイルが存在すると上書きを避けるため処理が中断されてしまいます。なお、対話式モードの場合は、「-w」が指定されていないと、ファイルを上書きしてよいかどうか問い合わせます。

eDirectory のフル/インクリメンタルバックアップの都度、すぐにファイルシステムのバックアップを取っているのであれば、前回のバックアップファイルはテープに保存されているはずですが、したがって上書きしても問題ありません。

- ◆ この例では安全でないポートを指定している (-p 8008) ので、それを考慮して接続するよう、「-n」オプションも指定しています。

## Windows 用のバッチファイル例

```
java -cp c:\novell\nds\embox\emBoxClient.jar embox -s myserver -p 8008 -u
admin.myorg -w mypassword -n -t backup.backup -b -f c:\backup\backup.bak -u
c:\backup\includes\includefile.txt -l c:\backup\backup.log -e -t -w
```

この例には次のオプションが指定されています。

- ◆ フルバックアップを取る指定 (-b)。
- ◆ インクルードファイルの指定 (-u)。これは必要な場合のみ指定してください。データベース以外にもファイルをバックアップしたい場合に使います。インクルードファイルはあらかじめ用意しておいてください。
- ◆ NCI セキュリティファイル (-e)、ストリームファイル (-t) もバックアップする指定。NCI ファイルは常にバックアップするようお勧めします。
- ◆ 同じ名前のバックアップファイルがあれば上書きする指定 (-w)。

**重要：**同じバッチファイルを繰り返し実行するなど、同じ名前のバックアップファイルがある場合、「-w」を指定しないと正常にバックアップされません。

バッチモードでは、「-w」の指定がなければ、同じ名前のファイルが存在すると上書きを避けるため処理が中断されてしまいます。なお、対話式モードの場合は、「-w」が指定されていないと、ファイルを上書きしてよいかどうか問い合わせます。

eDirectory のフル/インクリメンタルバックアップの都度、すぐにファイルシステムのバックアップを取っているのであれば、前回のバックアップファイルはテープに保存されているはずですが、したがって上書きしても問題ありません。

- ◆ この例では安全でないポートを指定している (-p 8008) ので、それを考慮して接続するよう、「-n」オプションも指定しています。

**注：**NetWare 用のバッチファイル例に含まれていた「-ns」または「ac」オプションは、NetWare プラットフォームでのみ指定するべきものです。Windows や UNIX では指定しないでください。

## eMBox クライアントによるロールフォワードログの設定

eMBox クライアントを使って、ロールフォワードログに関する設定を変更することができます。次の設定ができます。

- ◆ 現在の設定の確認
- ◆ ロールフォワードログ機能の有効/無効の切り替え  
レプリカリングに属するサーバは、ロールフォワードログ機能を有効にしておく必要があります。バックアップファイルがあっても、ロールフォワードログがなければ復元後の検証処理に失敗し、データベースを開けないこととなります。
- ◆ ロールフォワードログの保存先ディレクトリの変更
- ◆ ロールフォワードログのファイルサイズの最小値、最大値の設定
- ◆ 現在使用中のログ、既に書き出しを終えた最新のログの判別
- ◆ ストリームファイルをロールフォワードログに含めるかどうかの切り替え

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。

## 前提条件

- 設定の変更処理を起動するコンピュータに、ファイル `eMBoxClient.jar` があることを確認してください。

このファイルは、eDirectory の一部としてサーバにインストールされます。それ以外にも、Sun JVM 1.3.1 が動作する環境であれば、`eMBoxClient.jar` をコピーして動かすことができます。複数サーバ環境でも、ファイアウォール越しのアクセスが可能であれば1台のコンピュータから作業できます。詳細については、[479 ページの「eMBox コマンドラインクライアントの使用」](#)を参照してください。

- コマンドラインオプションについては、[423 ページの「バックアップ / 復元のコマンドラインオプション」](#)を参照してください。

## 操作手順

- 1 eMBox クライアントを対話式モードで起動します。

- ◆ NetWare、UNIX の場合：コマンドラインから「`edirutil -i`」と入力します。
- ◆ Windows の場合：「`drive\novell\nds\embox\edirutil.exe -i`」を実行します。

`edirutil` ファイルは、eMBox クライアントを実行するためのショートカットです。Java の実行形式ファイルと、eMBox クライアントのインストール先ディレクトリがパラメータとして記述されているほか、NetWare の場合は「`-ns`」オプションもついています。(Java 実行ファイルの場所は、[481 ページの「ワークステーションで eMBox クライアントを実行する」](#)で示しているように手動で入力することもできます)。

正常に起動されると、「eMBox Client>」というプロンプトが現れます。

- 2 ロールフォワードログの設定を行うサーバにログインします。次のように入力してください。

```
login -s server_name_or_IP_address -p port_number  
-u username.context -w password
```

たとえば Windows の場合、次のようになります。

```
login -s 151.155.111.1 -p 8009 -u admin.mycompany -w mypassword
```

セキュア接続が確立できないというエラーが表示される場合は、[487 ページの「eMBox クライアントを使用してセキュア接続を確立する」](#)に表示されている JSSE ファイルがシステム上にない可能性があります。

指定するポート番号が分からない場合は、[488 ページの「eDirectory ポート番号を確認する」](#)を参照してください。

eMBox クライアントはログインが成功したかどうかを表示します。

- 3 (オプション) 次のように入力して、現在の設定を確認します。

```
getconfig
```

オプション指定は必要ありません。

たとえば次のように表示されます。

```
Roll forward log status OFF  
Stream file logging status OFF  
Current roll forward log directory voll:/rfl/nds.rfl  
Minimum roll forward log size (bytes) 104857600  
Maximum roll forward log size (bytes) 4294705152  
Last roll forward log not used 00000000.log  
Current roll forward log 00000001.log  
*** END ***
```

- 4 setconfig コマンドで設定を変更します。次の形式で入力してください。

```
setconfig [-L|-l] [-T|-t] -r path_to_roll-forward_logs -n minimum_file_size -m maximum_file_size
```

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。たとえば NetWare の場合、次のようになります。

```
setconfig -L -r rflvolume:\logs
```

これは、ロールフォワードログ機能を有効にし (-L スイッチ)、その保存先を「rflvolume:\logs」以下と指定するコマンドです。ロールフォワードログ専用のディスクパーティション / ボリュームを用意するのが最善です。こうしておけば、ディスク容量やアクセス権を監視しやすくなります。ただしこの例では、ストリームファイルのログ機能は有効にしていません。

**警告:** ロールフォワードログ機能を有効にしたら、デフォルトの保存先を使用しないでください。障害対策のためには、eDirectory とは別のディスクパーティション / ボリューム、別の記憶デバイスを指定してください。ロールフォワードログディレクトリは、バックアップ環境設定を変更するサーバ上である必要があります。

**重要:** ロールフォワードログ機能を有効にする場合、ログを保存するボリュームのディスク容量を常に監視してください。これを怠ると、ログの容量は増える一方なので、ディスクパーティション / ボリュームがあふれてしまう恐れがあります。ディスク容量が不足してロールフォワードログを作成できない場合は、eDirectory はそのサーバに対して応答しなくなります。書き出しが終わったロールフォワードログは、定期的にバックアップし、サーバから削除するようお勧めします。400 ページの「**ロールフォワードログのバックアップと削除**」を参照してください。

- 5 サーバからログアウトするには、次のコマンドを入力します。

```
logout
```

- 6 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

## eMBox クライアントによるバックアップファイルの復元作業

eMBox クライアントを使ってバックアップファイルに保存されたデータから eDirectory データベースを復元する手順を解説します。手動あるいはバッチ方式で残しておいたバックアップファイルから、データを復元できます。処理結果は所定のログファイルに記録されます。

eMBox クライアントを使えば、iManager では実現できない高度な復元機能も実行できます。詳しくは 423 ページの「**バックアップ / 復元のコマンドラインオプション**」に、「**restore**」および「**restadv**」として解説します。

### 前提条件

- 復元処理を起動するコンピュータに、ファイル eMBoxClient.jar があることを確認してください。

このファイルは、eDirectory の一部としてサーバにインストールされます。それ以外にも、Sun JVM 1.3.1 が動作する環境であれば、eMBoxClient.jar をコピーして動かすことができます。複数サーバ環境でも、ファイアウォール越しのアクセスが可能であれば 1 台のコンピュータから作業できます。詳細については、479 ページの「**eMBox コマンドラインクライアントの使用**」を参照してください。



- 必要なバックアップファイルをすべて、復元対象サーバ上の、適当なディレクトリに集めておく必要があります。

401 ページの「復元処理の準備」および 403 ページの「復元に必要なバックアップファイルの収集」を参照してください。
- 復元対象サーバに eDirectory をインストールし、稼働させておいてください。

たとえば記憶デバイスの障害の場合、デバイスを交換し、改めて eDirectory をインストールすることになります。故障したサーバごと交換する、あるいは単に新しいサーバに移行する場合は、新しいサーバにオペレーティングシステムをインストールした上で、eDirectory も準備します。
- コマンドラインオプションについては、423 ページの「バックアップ / 復元のコマンドラインオプション」を参照してください。
- 復元処理の詳細については、388 ページの「Backup eMTool による復元作業の概要」を参照してください。
- (NetWareのみ)ファイルシステムデータおよびeDirectoryを復元する際は、ファイルシステム権利を元どおりに戻せるよう、正しい手順で作業してください。eDirectoryを先に復元し、その後でファイルシステムを復元してください。この手順の詳細については、396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」を参照してください。

## 操作手順

eMBox クライアントを使って eDirectory データベースを復元する手順を示します。

- 1 必要なバックアップファイルを集めておきます。詳しくは 401 ページの「復元処理の準備」を参照してください。
- 2 eMBox クライアントを対話式モードで起動します。
  - NetWare、UNIX の場合：コマンドラインから「**edirutil -i**」と入力します。
  - Windows の場合：「**drive\novell\nds\embox\edirutil.exe -i**」を実行します。

edirutil ファイルは、eMBox クライアントを実行するためのショートカットです。Java の実行形式ファイルと、eMBox クライアントのインストール先ディレクトリがパラメータとして記述されているほか、NetWare の場合は「-ns」オプションもついています。(Java 実行ファイルの場所は、481 ページの「ワークステーションで eMBox クライアントを実行する」で示しているように手動で入力することもできます)。

正常に起動されると、「eMBox Client>」というプロンプトが現れます。

- 3 復元の対象サーバにログインします。次のように入力してください。

```
login -s server_name_or_IP_address -p port_number  
-u username.context -w password
```

たとえば Windows の場合、次のようになります。

```
login -s 151.155.111.1 -p 8009 -u admin.mycompany -w mypassword
```

セキュア接続が確立できないというエラーが表示される場合は、487 ページの「eMBox クライアントを使用してセキュア接続を確立する」に表示されている JSSE ファイルがシステム上にない可能性があります。

指定するポート番号が分からない場合は、488 ページの「eDirectory ポート番号を確認する」を参照してください。

eMBox クライアントはログインが成功したかどうかを表示します。

- 4 eMBoxクライアントのプロンプトが出たら、次の形式で復元コマンドを入力します。

```
restore -r -a -o -f full_backup_path_and_filename  
-d roll-forward_log_location -l restore_log_path_and_filename -e
```

各スイッチの間にはスペースが必要です。スイッチの順序は重要ではありません。「-r」オプションを指定すれば eDirectory データベース自身、指定しなければそれ以外のファイルのみが復元の対象となります。復元処理の終了後、データベースをアクティブにし、オープンしたい場合は、「-a」および「-o」を指定してください。NICI 機能を使っている場合は NICI ファイルも復元 (-e) します。そうしないと暗号化されたデータを読むことができません。

ロールフォワードログを使って復元する場合は、そのフルパスを指定しなければなりません。これには、eDirectory が自動的に追加するディレクトリ名 (通常は「nds.rfl」) も含まれます。詳しくは 399 ページの「ロールフォワードログの保存先」を参照してください。

次に例を示します。

```
restore -r -a -o -f sys:/backup/nds.bak -d voll1:rflmdir/nds.rfl -l sys:/backups/backup.log -e
```

この例では、データベースを復元 (-r) し、その検証が正常終了したらアクティブにし (-a)、オープンする (-o) よう指定しています。「-f」オプションでフルバックアップファイル、「-d」オプションでロールフォワードログの保存先を指定します。また、復元処理の結果を記録するログファイルを、「-l」オプションで指定しています。さらに、NICI ファイルも復元します (-e)。

これによりフルバックアップファイルからの復元処理が実行され、次にインクリメンタルバックアップファイルの指定を求めるプロンプトが現れます。

- 5 (状況によって実行) インクリメンタルバックアップファイルから復元する場合は、プロンプトに応じて順次、そのパスとファイル名を入力します。

プロンプトには次に指定するべきファイルの ID が表示されます。これはインクリメンタルバックアップファイルのヘッダに記載されているものです。

バックアップ処理が正常終了すれば、その旨の表示が現れます。

- 6 (状況によって実行) 復元処理に失敗した場合は、ログファイルでエラーの原因を確認してください。

復元後の検証に失敗した場合の対処については、432 ページの「復元後の検証処理に失敗した場合の対処方法」を参照してください。

注：レプリカリング中に eDirectory 8.5 より前のバージョンが稼動しているサーバがある場合、復元処理は失敗します。エラーコードは -666、すなわち「DS バージョンの不整合」となります。この場合の対処方法については、395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」を参照してください。

- 7 サーバからログアウトするには、次のコマンドを入力します。

```
logout
```

- 8 eMBox クライアントを終了するには、次のコマンドを入力します。

```
exit
```

- 9 (状況によって実行) NICI セキュリティファイルを復元した場合は、NICI を再初期化するため、サーバを再起動します。

- 10 ここでサーバが通常どおり要求に応答することを確認しておきます。

- 11** (状況によって実行) このサーバでロールフォワードログ機能を使うためには、改めて有効に切り替え、障害対策のための書き出し先も設定し直して、ロールフォワードログの環境設定を再作成する必要があります。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

この手順が必要となるのは、復元処理の過程で、ロールフォワードログに関する設定はデフォルトに戻るためです。つまり、ロールフォワードログ機能は無効となり、保存先もデフォルトの場所になるからです。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#)を参照してください。

以上で復元作業が終了しました。NICI の再初期化も済んでいるので、暗号化された情報にもアクセスできます。ロールフォワードログ機能を使用する場合は、今後の障害に備えるため、再びこの機能を有効にし、フルバックアップを取っておいてください。

## バックアップ / 復元のコマンドラインオプション

eDirectory Backup eMTool のコマンドラインオプションを、[backup](#)、[restore](#)、[restadv](#)、[getconfig](#)、[setconfig](#)、[cancel](#) の各コマンドについて解説します。

オプションはどのような順序で指定しても構いません。各オプション間は空白で区切ってください。

オプション	説明
<b>backup</b>	データベースおよび関連ファイルのバックアップ
<b>-f file_name</b>	(必須) バックアップファイルの名前とパス  Backup eMTool で作成するバックアップ先ファイル名とパスを指定します。バックアップ対象サーバ上のローカルファイルを指定してください。たとえば「 <code>backup -f vol1:\backup\ndsbak.bak</code> 」と指定すると、ファイル <code>vol1:\backup\ndsbak.bak</code> にバックアップされます。
<b>-l file_name</b>	(必須) ログファイルの名前とパス  バックアップ処理の結果を出力するログファイルを指定します。
<b>-b</b>	(オプション) フルバックアップを実行  eDirectory データベースのフルバックアップを取ります。これがデフォルトの動作で、「 <code>-i</code> 」も「 <code>-c</code> 」も指定しなければフルバックアップになります。
<b>-i</b>	(オプション) インクリメンタルバックアップを実行  eDirectory データベースのインクリメンタルバックアップを取ります。最後に実施したフル / インクリメンタルバックアップ以降、変化した部分のみをバックアップします。

オプション	説明
-e	<p>(オプション)セキュリティファイルもバックアップ</p> <p>NICI セキュリティファイルもバックアップの対象とします。バックアップの対象としてインクルードファイルに記載したものと同様の形式でバックアップされますが、タイプタグは「user」ではなく「nici」となります。</p> <p>サーバに暗号化されたデータがある場合、復号するにはシステムがバックアップされたときと同じ NICI ファイルが必要です。それらのファイルはローカルデータベースと連携しています。</p> <p><b>警告</b> : NICI 機能を使っている場合は、NICI ファイルも確実にバックアップするようにしてください。これがないと、復元に成功しても、暗号化されたデータを読むことはできなくなります。</p>
-t	<p>(オプション)ストリームファイルもバックアップ</p> <p>eDirectory データベースをバックアップする際、ストリームファイルも含める指定です。</p>
-u <i>file_name</i>	<p>(オプション)インクルードファイル名およびパス</p> <p>バックアップ対象に追加するファイル名を列挙した、インクルードファイルを指定します。eDirectory データベースを復元する際に必要となる重要なファイルをいっしょにバックアップしたい場合、インクルードファイルに列挙しておきます。</p> <p>インクルードファイルには各ファイルのフルパスを記述し、末尾にセミコロン (;) を置いてください。たとえば autoexec.ncf および hosts を NetWare サーバのバックアップファイル対象として追加する場合、インクルードファイルは次のようになります。</p> <pre>sys:\system\autoexec.ncf;sys:\etc\hosts;</pre> <p>ファイルのリストにはスペースまたは改行 (ハードリターン) を使用しないでください。</p> <p>指定どおりバックアップされたことは、ログファイルを見るか、バックアップファイルのヘッダを見れば確認できます。(393 ページの「バックアップログファイルの書式」および 389 ページの「バックアップファイルのヘッダ書式」を参照してください)。</p> <p><b>警告</b> : バックアップファイルを開く場合でもヘッダを確認するだけにとどめ、保存や変更はしないようにしてください。ファイルの一部が切り捨てられてしまうことがあります。ほとんどのアプリケーションではバイナリデータを正確に保存することはできません。</p>

オプション	説明
-s <i>file_size</i>	<p>(オプション)バックアップファイルの最大容量 (バイト単位)。</p> <p>バックアップファイルの最大容量を、バイト単位で指定します。バックアップファイルを保存する記憶媒体に容量制限がある場合、このオプションで最大容量を指定するとよいでしょう。</p> <p>最大容量に達すると新しいバックアップファイルが生成されます。ファイル名の末尾に、5桁の16進数値を追加した名前になります。この拡張の数字は、新規ファイルが作成されるたびに1ずつ増加します。</p> <p>たとえば「<code>backup -f vol1:/backup/mydib.bak -s 1000000</code>」というコマンドの場合、バックアップファイルの最大容量は1MBになります。データベースが3.5MBあったとすれば、最終的に次の4つのバックアップファイルができます。</p> <pre>vol1:/backup/mydib.bak (容量は 1MB) vol1:/backup/mydib.bak.00001 (容量は 1MB) vol1:/backup/mydib.bak.00002 (容量は 1MB) vol1:/backup/mydib.bak.00003 (容量は 0.5MB)</pre> <p>指定できるサイズの下限は約500KBです。バックアップで作成されるファイル数によって、最初のファイルが大きくなる場合があります。</p> <p>最初のバックアップファイルには「<code>number_of_files</code>」タグが追加されます。これはバックアップファイルの総数を表します。上記の例では4となります。さらに、各ファイルのヘッダに、「<code>backup_file</code>」属性が追加されます。これは本来のバックアップファイル名を表します (詳細については、<a href="#">389 ページの「バックアップファイルのヘッダ書式」</a>を参照してください)。</p> <p>上記の4つのバックアップファイルを使って復元する場合、コマンドは次のようになります。</p> <pre>restore -f vol1:/backup/mydib.bak -l <i>log_file_path_and_filename</i></pre> <p>ファイルが複数に分かれていることはBackup eMToolによって自動的に認識され、同じディレクトリ内にある、上記の名前のファイルが検索されます。</p> <p><b>ヒント:</b> サードパーティ製の圧縮ツールを使えば、バックアップファイルの容量を小さくすることができます。80%程度は圧縮できます。</p>
-w	<p>(オプション)同名のバックアップファイルがあれば上書き</p> <p>「-f」オプションで指定されたのと同じ名前のバックアップファイルがあれば、上書きします。この指定がない場合で同名のファイルが存在すると、対話式モードであれば、Backup eMToolは上書きしてよいかどうか確認を求めます。バッチモードでは、「-w」の指定がなければ、同名のファイルが存在すると上書きを避けるため処理が中断されてしまいます。</p> <p>eDirectoryのフル/インクリメンタルバックアップの都度、すぐにファイルシステムのバックアップを取っているのであれば、前回のバックアップファイルはテープに保存されているはずで、したがって上書きしても問題ありません。</p> <p><b>重要:</b> バッチファイルを使って無人バックアップを行う場合、このオプションを指定してください。同じバッチファイルを繰り返し実行するなど、同名のバックアップファイルがある場合、「-w」を指定しないと正常にバックアップされません。</p> <p>バッチモードでは、「-w」の指定がなければ、同名のファイルが存在すると上書きを避けるため処理が中断されてしまいます。なお、対話式モードの場合は、「-w」が指定されていないと、ファイルを上書きしてよいかどうか問い合わせます。</p>

オプション	説明
-c	<p>(オプション) コールドバックアップを実行</p> <p>フルバックアップと同様ですが、いったんデータベースを停止してから実行します。「-o」または「-o -d」が指定されている場合を除き、処理終了後、データベースは再びオープンされます。</p>
-o	<p>(オプション) コールドバックアップ後、データベースを停止したままにする</p> <p>「-c」を指定した場合にのみ指定できます。コールドバックアップの終了後、データベースを停止したままにします。この機能は、ハードウェアをアップグレードする、あるいは新規サーバ(同じオペレーティングシステムが動作するもの)に移行する場合に有用です。詳しくは <a href="#">460 ページの「ハードウェアのアップグレードやサーバの交換」</a> を参照してください。</p>
-d	<p>(オプション) コールドバックアップ後、DS エージェントを無効にする</p> <p>「-c -o」を指定した場合にのみ指定できます。コールドバックアップ後、DS エージェントを無効にします。この機能は、ハードウェアをアップグレードする、あるいは新規サーバ(同じオペレーティングシステムが動作するもの)に移行する場合に有用です。詳しくは <a href="#">460 ページの「ハードウェアのアップグレードやサーバの交換」</a> を参照してください。</p> <p>擬似サーバの「login disabled」属性を設定することにより、DS エージェントを無効にします。その結果、eDirectory を起動しようとすると「-663」エラーが発生します。</p>
<b>restore</b>	<b>データベースおよび関連ファイルの復元</b>
-f <i>file_name</i>	<p>(必須) バックアップファイルの名前とパス</p> <p>復元に使うフルバックアップファイルを指定します。このファイルは復元対象サーバ上に置いておかなければなりません。たとえば「restore -f vol1:/backup/ndsbak.bak」と指定すると、ファイル vol1:/backup/ndsbak.bak から復元されます。</p> <p>複数のファイルに分かれている場合は、すべて同じディレクトリ内に集めておいてください。</p>
-l <i>file_name</i>	<p>(必須) ログファイルの名前とパス</p> <p>復元処理の結果を出力するログファイルを指定します。</p>
-r	<p>(オプション) DIB セットも復元</p> <p>eDirectory データベースを復元する指定です。</p> <p><b>警告:</b> このオプションを指定しなかった場合、eDirectory データベース自身は復元されません。NICI セキュリティファイル (-e) など、他に指定したファイルのみが復元の対象になります。</p>

オプション	説明
-d <i>dir_name</i>	<p>(オプション) ロールフォワードログのあるディレクトリ</p> <p>ロールフォワードログを集めたディレクトリを指定します。復元対象サーバ上のフルパスで指定してください。必要なロールフォワードログをすべて、作成時と同じファイル名にして、ひとつのディレクトリに集めておかなければなりません。</p> <p>バックアップファイルからの復元後、ロールフォワードログを使って、バックアップ時点以降の変更を反映させます。「-d」オプションの指定がなければ、バックアップ時にロールフォワードログ機能を有効にしても、Backup eMTool はログファイルを参照しません。</p> <p>最初に適用すべきロールフォワードログは、最新のバックアップファイルをテキストエディタで開き、「backup」タグの「current_log」属性を見れば確認できます。ここでいう最新のバックアップファイルとは、「-f」オプションで指定するフルバックアップファイルか、または復元処理で適用することになる最後のインクリメンタルバックアップファイルです。ヘッダに記述される属性について詳しくは、<a href="#">389 ページの「バックアップファイルのヘッダ書式」</a>を参照してください。</p> <p><b>警告:</b> バックアップファイルを開く場合でもヘッダを確認するだけにとどめ、保存や変更はしないようにしてください。ファイルの一部が切り捨てられてしまうことがあります。ほとんどのアプリケーションではバイナリデータを正確に保存することはできません。</p>
-e	<p>(オプション) セキュリティファイルも復元</p> <p>NICI セキュリティファイルをバックアップしていた場合、それも復元します。</p> <p><b>警告:</b> NICI 機能を使っている場合は、NICI ファイルもバックアップされているはずなので、これも復元してください。そうしないと、復元に成功しても、暗号化されたデータを読むことはできなくなります。</p>
-u	<p>(オプション) インクルードファイルに列挙されたファイルも復元</p> <p>データベースに追加する形でバックアップしていたファイルも復元します。</p> <p>バックアップの過程で、データベース以外にもバックアップが必要なファイルを列挙したファイルを作成し、インクルードファイルとして指定することもできます。しかしその場合でも、「-u」オプションで指定しなければ復元されません。</p>
-a	<p>(オプション) 検証後、DIB をアクティブにする指定。</p> <p>復元後の検証処理が正常終了したら、データベース名を RST から NDS に変更します。この処理については、<a href="#">388 ページの「Backup eMTool による復元作業の概要」</a>を参照してください。</p>
-o	<p>(オプション) 処理終了後、データベースをオープンする</p> <p>復元作業が終了したら、自動的にデータベースをオープンする指定です。検証処理が正常終了すれば、データベースが自動的に開きます。失敗した場合は、復元前のデータベースが開きます。この処理については、<a href="#">388 ページの「Backup eMTool による復元作業の概要」</a>を参照してください。</p>

オプション	説明
-n	<p>(オプション) 復元後にデータベースを検証しない</p> <p>復元後の検証処理を省略して Backup eMTool で復元します。このサーバの遷移ベクトルをレプリカリングに属する他のサーバと比較する、という検証処理を行いません。遷移ベクトルについて詳しくは、<a href="#">395 ページの「遷移ベクトルと復元後の検証処理」</a>を参照してください。他のオプションで明示的に指定されていない限り、RST から NDS への改名もしません。</p> <p><b>重要:</b> Novell の担当者から指示を受けた場合を除き、このオプションはお勧めできません。</p>
-v	<p>(オプション) 上書きして復元</p> <p>検証処理を行うことなく、データベース名を RST から NDS に変更します。</p> <p><b>重要:</b> Novell の担当者から指示を受けた場合を除き、このオプションはお勧めできません。</p>
-k	<p>(オプション) データベースのロックを解除</p> <p>NDS データベースのロックを解除します。</p>
restadv	<p><b>高度な復元機能。注意:</b> これを実行する際は、DS エージェントがクローズされます。</p>
-l <i>file_name</i>	<p>(必須) ログファイルの名前とパス</p> <p>復元処理の結果を出力するログファイルを指定します。</p>
-o	<p>(オプション) 処理終了後、データベースをオープンする</p> <p>復元作業が終了したら、自動的にデータベースをオープンする指定です。検証処理が正常終了すれば、データベースが自動的に開きます。失敗した場合は、復元前のデータベースが開きます。この処理については、<a href="#">388 ページの「Backup eMTool による復元作業の概要」</a>を参照してください。</p>
-n	<p>(オプション) 前に失敗した復元の検証処理を起動</p> <p>前に復元して検証に失敗した RST データベースを再度検証します。</p>
-m	<p>(オプション) 復元された DIB ファイルの削除</p> <p>RST データベースが存在すれば削除します。</p>
-v	<p>(オプション) 上書きして復元</p> <p>検証処理を行うことなく、データベース名を RST から NDS に変更します。</p> <p><b>重要:</b> Novell の担当者から指示を受けた場合を除き、このオプションはお勧めできません。</p>
-k	<p>(オプション) データベースのロックを解除</p> <p>NDS データベースのロックを解除します。</p>



オプション	説明
getconfig	ロールフォワードログに関する現在の設定を表示。
	<p>指定できるオプションはありません。</p> <p>現在の設定を表示します。たとえばロールフォワードログ機能が無効になっている場合、getconfig コマンドでは次のような情報が表示されます。</p> <pre data-bbox="579 350 1330 576"> Roll forward log status OFF Stream file logging status OFF Current roll forward log directory voll:/rfl/nds.rfl Minimum roll forward log size (bytes) 104857600 Maximum roll forward log size (bytes) 4294705152 Last roll forward log not used 00000000.log Current roll forward log 00000001.log *** END *** </pre>
setconfig	ロールフォワードログに関する設定の変更。
-L	<p>(オプション) ロールフォワードログ機能の有効化。</p> <p>ロールフォワードログ機能を有効にします。(デフォルトでは無効)。この機能を有効にしておけば、停止する直前の状態にまでサーバを復元できるようになります。無効のままであれば、最後のフル/インクリメンタルバックアップ時点までしか復元できません。</p> <p>レプリカリングに属するサーバについては、ロールフォワードログ機能を有効にして、他のサーバとの同期状態も復元できるようにしてください。</p> <p>ただし管理者にとっては、監視しなければならない対象が増えます。これを怠ると、ログの容量は増える一方なので、ディスクパーティション/ボリュームがあふれてしまう恐れがあります。ディスク容量が不足してロールフォワードログを作成できない場合は、eDirectory はそのサーバに対して応答しなくなります。定期的なバックアップを取り、使わなくなったログは削除する必要があります。<a href="#">400 ページの「ロールフォワードログのバックアップと削除」</a>を参照してください。</p> <p>詳細については、<a href="#">397 ページの「ロールフォワードログを使用する」</a>を参照してください。</p>
-l	<p>(オプション) ロールフォワードログ機能の無効化。</p> <p>ロールフォワードログ機能を無効にします(デフォルトでは無効)。データベースでは連続したログを保存していくのをやめ、現在のファイルに上書きしていくようになります。ロールフォワードログ作成がオフの場合、最後にフル/インクリメンタルバックアップを実行した時点までしかデータベースを復元できません。</p> <p>誤って無効にしてしまった場合、ただちに有効にすると同時に、今障害が起こっても復元できるよう、改めてデータベースのバックアップを取ってください。</p> <p>詳細については、<a href="#">397 ページの「ロールフォワードログを使用する」</a>を参照してください。</p>

オプション	説明
-T	<p>(オプション) ストリームファイルのログ出力開始</p> <p>(ロールフォワードログ機能が有効な場合のみ) ストリームファイルが更新された場合、その全体をロールフォワードログにコピーするようになります。ストリームファイルとは、ログインスクリプトなど、データベースに関する追加の情報ファイルのことです。</p> <p>ただしストリームファイルを記録すると、ディスクの空き容量が急速に減少します。ログの出力先ディスクパーティション/ボリュームの空き容量を、常に監視するようにしてください。ディスク容量が不足してロールフォワードログを作成できない場合は、eDirectory はそのサーバに対して応答しなくなります。</p>
-t	<p>(オプション) ストリームファイルのログ出力停止</p> <p>ストリームファイルが更新されても、その全体をロールフォワードログにコピーしないようになります。この場合でも、フル/インクリメンタルバックアップの際には、ストリームファイルもバックアップできます。ストリームファイルを頻繁に更新しないのであれば、それでも充分でしょう。</p> <p>ストリームファイルを記録しないと、ログファイルの容量が急速に増えるのを抑えることにもなります。</p>
-r <i>dir_name</i>	<p>(オプション) ロールフォワードログの出力先ディレクトリの設定。</p> <p>ロールフォワードログの出力先ディレクトリを指定します。たとえば「setconfig -r vol2:\rfl」というコマンドを実行すると、vol2:\rfl 以下にディレクトリが作成され、その下にログファイルができるようになります。</p> <p>このディレクトリ名は現在の eDirectory データベース名に基づいて決まります。通常はデータベース名が「NDS」なので、ログ保存先ディレクトリは「vol2:\rfl\nds.rfl」となります。ここでデータベース名を「ND1」に変更すると、保存先もこれに合わせて「vol2:\rfl\nd1.rfl」以下に変わります。</p> <p>現在の保存先設定は getconfig コマンドで確認できます。</p> <p>保存先の設定を変えるとその時点で新しいディレクトリができますが、ログファイルは実際にトランザクションが発生するまで作成されません。</p> <p><b>重要:</b> バックアップツールでは、ログの保存先ディレクトリが変わったことを認識できません。データベースを復元する際には、最後のバックアップ以降のロールフォワードログをすべて、ひとつのディレクトリに集めておく必要があります。</p> <p>詳細については、<a href="#">397 ページの「ロールフォワードログを使用する」</a>を参照してください。</p>
-n <i>file_size</i>	<p>(オプション) ロールフォワードログの最小容量の設定</p> <p>ロールフォワードログの最小容量をバイト単位で指定します。この容量に達した後、実行中のトランザクションが終了すると、ログ出力先が新しいファイルに切り替わります。</p>
-m <i>file_size</i>	<p>(オプション) ロールフォワードログの最大容量の設定</p> <p>ロールフォワードログの最大容量をバイト単位で指定します。この上限に達してもトランザクションが進行中の場合は、トランザクションは次のファイルに続けて記録されます。この設定は最小サイズの設定より常に大きくする必要があります。</p>

オプション	説明
-s	(オプション) ログ出力先ファイルの強制切り替え  実行中のトランザクションが終了した時点で、ログ出力先を新しいファイルに切り替えます。次のトランザクション開始時に新しいファイルが作成されます。
cancel	バックアップ/復元処理を取り消します。  指定できるオプションはありません。

## サーバ固有情報のバックアップに関する変更事項 (NetWare のみ)

NetWare 用インストールでは、多くの場合、サーバ固有情報もバックアップするようになっています。eDirectory 8.6 以降、eDirectory スキーマの構造が変更になりました。eDirectory 8.7 でも、さらに多くの変更が施されています。その結果、ファイルシステム TSA やサードパーティ製バックアップツールを使ってサーバ固有情報をバックアップしていた場合、そのままでは使えなくなりました。代わりに追加された Backup eMTool の「ホットバックアップ」機能を iManager から呼び出すか、または eMBox クライアントで実行するようにしてください。ファイルシステム TSA を使ってサーバ固有情報をバックアップする機能は、現在のバージョンからは削除されました。eDirectory 8.7.3 では、該当機能が「ホットバックアップ」に組み込まれています。従来と同様、ファイルシステム TSA では、dsbacker.nlm を呼び出してバックアップファイルを作成しています。しかし現在は、dsbacker.nlm が内部的に backupcr.nlm を起動し、これがさらに Backup eMTool の機能を呼び出すという形で実装されています。

NetWare および eDirectory のバージョンに応じて、次のようにバックアップ/復元処理を行うようお勧めします。

eDirectory のバージョン	NetWare のバージョン	推奨するバックアップ/復元方法
8.6 以前	すべてのバージョン	ファイルシステム TSA を使ってサーバ固有情報 (SSI) を復元する場合： <ul style="list-style-type: none"> <li>◆ 停止したサーバに関連づけられているボリュームオブジェクト、サーバオブジェクトを削除しないでください。</li> <li>◆ 具体的な手順については Novell の担当者にお問い合わせください。</li> </ul>
8.7	5.1 & 6.0	バックアップ/復元には Backup eMTool のみを使用。  (ファイルシステム TSA でバックアップしても復元できません)。
8.7.1 以降	5.1	バックアップ/復元には Backup eMTool のみを使用。  (ファイルシステム TSA で SSI をバックアップしても復元できません)。
8.7.1 以降	6.0 (SP3 を適用済み)  (eDirectory 8.7.1 を動かすには SP3 が必要)	Backup eMTool、ファイルシステム TSA、サードパーティ製ツールのいずれかを使用。復元処理は Backup eMTool で実行できます。

NetWare 6.0 上で eDirectory 8.7.1 を動かす場合、サーバ固有情報で大きく変わったのは次の事項です。

- ◆ **ファイル長の増大:** 以前の SSI バックアップでは、データベースの一部しか対象になりませんでした。新バージョンでは、サーバ上のディレクトリオブジェクトに関する情報をすべてバックアップするようになったため、ファイル長も増大し、データベースそのものとはほぼ同じ容量になります。
- ◆ **ユーザ定義ファイルの場所:** 以前は sys:system ディレクトリにファイル servedata.nds が生成されるだけでした。容量が小さかったため、テープにコピーするまでこのディレクトリに置いておいても問題が生じなかったのです。eDirectory 8.7.3 では、ファイルシステム TSA でデータベースのフルバックアップが可能になりました。生成されるファイルは 3 つに増えています。そのうち ssiback.bak の保存位置は、ユーザが設定できます。

ファイル	説明	位置
ssiback.bak	Backup eMTool で生成されるフル「ホットバックアップ」ファイルと同じです。 <b>385 ページの「eDirectory Backup eMTool について」</b> を参照してください。	ユーザ定義可能。デフォルトでは sys:system になっています。  ファイル容量が大きいので、sys: 以外のボリュームに変更するようお勧めします。
ssiback.ini	ファイル ssiback.bak を書き出すパスを記述したテキストファイル。デフォルトでは「sys:system」となっています。  次に例を示します。  vol1:/backups/ssibackup.bak	sys:system
ssiback.log	前回までのバックアップ状況を分かりやすく出力したログファイルです。このログファイルにはすべてのバックアップ履歴、バックアップの開始および終了時刻、およびバックアッププロセス中に発生した考えられるエラーについての情報が含まれます。	sys:system

- ◆ **Backup eMTool による復元処理:** サーバ固有情報は Backup eMTool でしか復元できません。

## 復元後の検証処理に失敗した場合の対処方法

復元処理の一環として検証を行います。復元された eDirectory データベースと、レプリカリングに属する他のサーバとの間で、遷移ベクトルを比較するというものです。復元処理の詳細については、**388 ページの「Backup eMTool による復元作業の概要」**および **395 ページの「遷移ベクトルと復元後の検証処理」**を参照してください。

遷移ベクトルが合致しなければ、検証に失敗したことになります。これは一般に、復元処理に使ったファイルのデータが不足していたことを表します。たとえば次の状況が考えられます。

- ◆ 最後にバックアップを実施した後、ロールフォワードログ機能を有効にしていなかった場合。

- ◆ ロールフォワードログを取っていたのに、復元の際これを使わなかった場合。
- ◆ ロールフォワードログが不足していた場合。

**注:** このほか、レプリカリング中に、8.5 より古い eDirectory が稼動するサーバがある場合にも、検証に失敗します。この場合の対処方法については、[395 ページの「復元後の検証については eDirectory 8.5 以降のみで互換性がある」](#)を参照してください。

デフォルトでは、復元した eDirectory データベースが他のレプリカと整合が取れていない場合、そのままではオープンできません。

バックアップファイルやロールフォワードログの指定を単に忘れてただけであれば、正しく指定してもう一度復元処理を実行すればよいはずですが、次回に復元が成功すれば、検証に成功し、データベースがオープンされるでしょう。

必要なバックアップファイルやロールフォワードログが揃っていない場合は、以下の手順でサーバを復旧してください。検証に失敗したときの復旧手順の概要を説明します。

- ◆ サーバの識別情報やファイルシステム権利は、バックアップファイルなどが不足していても復旧できるはずですが。
- ◆ バックアップからレプリカを復元できない場合でも、サーバとしては動作します。新しいレプリカが追加されたものとして扱われ、他のサーバにも影響を及ぼしてしまうのです。したがって、レプリカリングからいったんサーバを外し、高度な復元機能や DSRepair ツールを使って元の状態に復旧してから、改めてレプリカリングに追加してください。
- ◆ このサーバにしかデータがない、すなわち他のサーバにレプリカが作られていないパーティションについては、残念ながら復元できません。

このセクションの説明により、検証に失敗した後、サーバの識別情報やファイルシステム権利を復旧し、レプリカリングからいったん外し、再び追加します。この手順でレプリケーションが終了すれば、サーバは元どおりに機能するようになるはずですが、ただしレプリカを作っていないために復元できなかったパーティションを除きます。

まず、[433 ページの「レプリカリングをクリーンアップする」](#)に従って作業してください。それが終了したら [435 ページの「サーバの復旧とレプリカの再追加」](#)に進みます。

## レプリカリングをクリーンアップする

この手順では、次の方法について説明します。

- ◆ **マスタレプリカの再割り当て。** 障害が発生したサーバが、あるパーティションのマスタレプリカを保持していた場合は、DSRepair を使って、レプリカリストに属する他のサーバ上のレプリカを、マスタとして扱うよう指定します。
- ◆ **障害が発生したサーバに対するレプリカリストの参照の削除。** 障害が発生したサーバを含むレプリカリングのメンバーである各サーバに対して、障害が発生したサーバが利用できなくなったことを通知する必要があります。

### 前提条件

- eDirectory そのものは、当該サーバに正常にインストールされているものとします。
- 復元を試み、検証処理で失敗したことが前提です。
- NDSデータベースは稼動しており、(復元処理により作成された)RSTデータベースも残っているものとします。
- どのパーティションのレプリカがこのサーバに保持されていたか、は分かっているものとします。このサーバのレプリカはバックアップファイルのヘッダ部に記録されています。

## 操作手順

レプリカリングをクリーンアップするには、次の操作を行います。

- 1 DSRRepairを起動します。復元対象サーバとレプリカを共有しているサーバのコンソールから、次のオプションを指定して起動してください。

- ◆ NetWare、Windows の場合：「-a」オプション。
- ◆ UNIX の場合：「-Ad」オプション。

上記のオプションについて詳しくは、[232 ページの「DSRepair のカスタムオプション」](#)を参照してください。

**警告：**「-a」オプション、「-Ad」オプションを指定して DSRRepair を実行すると、ツリー構造が損なわれることがあります。これらのオプションの詳細については、[Novell Support Web site, Solution 2938493 \(http://support.novell.com/servlet/tidfinder/2938493\)](http://support.novell.com/servlet/tidfinder/2938493) を参照してください。

- 2 レプリカ操作とパーティション操作の選択。
- 3 編集したいパーティションを選択します。このパーティションが属するレプリカリングから、障害の起こったサーバを外すこととなります。
- 4 [レプリカリングの表示] を選択して、このパーティションに関するレプリカを持つサーバのリストを表示します。
- 5 (状況によって実行) 当該サーバにマスタレプリカがあった場合、[このサーバを新しいマスタレプリカに設定] コマンドで、他のサーバにマスタを切り替えます。

この時点で、対象のレプリカリングは新しいマスタレプリカを保持しています。リングを構成するすべてのレプリカに対して、新しいマスタが存在することが通知されます。

- 6 マスタレプリカが確立されるまで待ちます。上記のレプリカがマスタになったことが他のサーバに認識されたことを確認してから続行します。
- 7 [レプリカリングの表示] 画面に戻ります。障害の起こったサーバを選択して、[レプリカリングからのサーバの削除] を実行してください。

DSRepair を起動する際に「-a」または「-Ad」(プラットフォームにより選択) を指定していなかった場合、このコマンドは表示されません。

**警告：** 障害の起こったサーバをマスタレプリカに設定したままで、このコマンドを実行しないでください。リング内のサーバリストを見れば確認できます。マスタレプリカであれば、[ステップ 5](#) を参照して、他のサーバにマスタを切り替えてから、当該サーバをレプリカリングから外します。

- 8 管理者としてログインします。
- 9 説明メッセージを読み、それに対する同意を入力して処理を続行します。
- 10 DSRRepair を終了します。

レプリカリングを構成するすべてのサーバに通知が行われます。

- 11 障害が発生したサーバを含んでいる各レプリカリングごとに、1 つのサーバ上で、この手順を繰り返します。

続いて、障害が生じたサーバ上に、新たにレプリカを構築します。[435 ページの「サーバの復旧とレプリカの再追加」](#)に進んでください。

## サーバの復旧とレプリカの再追加

レプリカ設定を「外部参照」側に書き替え、自分自身はレプリカリングに属していないものとして動作するようにします。その上でサーバからレプリカを削除すると、データベースのロックを解除できるようになります。

レプリカを削除すれば、レプリカをサーバに再追加する作業は終わりです。あとは自動的に、他のサーバから各レプリカの最新版を参照し、再追加していきます。各レプリカが再追加されたら、サーバは元と同じように機能するはずですが。

DSRepair を使用してレプリカを削除し、レプリケーション機能により再追加する手順を次に示します。

**1** 433 ページの「レプリカリングをクリーンアップする」が完了していることを確認します。

**2** eMBox クライアントの高度な復元機能を使って、上書き復元処理を行います。

**2a** eMBox クライアントを対話式モードで起動します。

- ◆ NetWare、UNIX の場合：コマンドラインから **edirutil -i** と入力します。
- ◆ Windows の場合：「**drive\novell\nds\embox\edirutil.exe -i**」を実行します。

edirutil ファイルは、eMBox クライアントを実行するためのショートカットです。Java の実行形式ファイルと、eMBox クライアントのインストール先ディレクトリがパラメータとして記述されているほか、NetWare の場合は「-ns」オプションもついています。(Java 実行ファイルの場所は、481 ページの「ワークステーションで eMBox クライアントを実行する」で示しているように手動で入力することもできます)。

正常に起動されると、「eMBox Client>」というプロンプトが現れます。

**2b** 復元の対象サーバにログインします。次のように入力してください。

```
login -s server_name_or_IP_address -p port_number -u
username.context -w password
```

たとえば Windows の場合、次のようになります。

```
login -s 151.155.111.1 -p 8008 -u admin.mycompany -w mypassword
```

セキュア接続が確立できないというエラーが表示される場合は、487 ページの「eMBox クライアントを使用してセキュア接続を確立する」に表示されている JSSE ファイルがシステム上にない可能性があります。

指定するポート番号が分からない場合は、488 ページの「eDirectory ポート番号を確認する」を参照してください。

eMBox クライアントはログインが成功したかどうかを表示します。

**2c** 上書き復元コマンドを、次のように実行します。併せてログファイル名も指定してください。

```
restadv -v -l logfilename
```

高度な復元機能により、RST データベース (復元したけれども検証に失敗したもの) の名前を NDS に変更します。ただしロックは解除しません。

- 3** サーバコンソールで、DSRepair の高度な復元機能により、レプリカ設定をすべて外部参照に切り替えます。

- ◆ NetWare の場合：「**dsrepair -xk2 -rd**」と入力します。
- ◆ Windows の場合： [スタート] > [設定] > [コントロールパネル] > [Novell eDirectory Services] の順にクリックします。dsrepair.dlm を起動します。[起動パラメータ] フィールドに、「**-xk2 -rd**」と入力します。[開始] をクリックします。
- ◆ UNIX の場合：「**ndsrepair -R -Ad -xk2**」と入力します。

オプション「-rd」、「-R」は、ローカルデータベースとレプリカを修復する旨の指定です。


**警告**：DSRepairによる高度な復元機能は、正しく使用しないとツリーが破損することがあります。これらのオプションの詳細については、[Novell Support Web site, Solution 2938493 \(http://support.novell.com/servlet/tidfinder/2938493\)](http://support.novell.com/servlet/tidfinder/2938493) を参照してください。

- 4** 修復が終了したら、ロックを解除してデータベースを開きます。次のように実行してください。

```
restadv -o -k -l logfilename
```

「-o」オプションはデータベースのオープン、「-k」オプションはロック解除を表します。

- 5** iManager を使って、修復されたサーバをレプリカリングに再追加します。

**5a** Novell iManager で、[役割およびタスク] ボタン  をクリックします。

**5b** [パーティションとレプリカ] > [レプリカビュー] の順にクリックします。

**5c** レプリカを作成したいパーティションの名前とコンテキストを指定し、[OK] ボタンを押します。

**5d** [レプリカの追加] をクリックします。

**5e** [サーバ名] フィールドの横にある [参照] ボタン  をクリックし、修復されたサーバを選択します。

**5f** レプリカのタイプを選択して、[OK] をクリックし、[完了] をクリックします。

**5g** このサーバが属していた各レプリカリングについて、上記の操作を繰り返します。

- 6** レプリケーション処理が終わるまでしばらく待ちます。

レプリカの状態が [新規] から [オン] に変われば、レプリケーション処理は終了です。これは iManager で確認できます。詳細については、[128 ページの「レプリカに関する情報を表示する」](#) を参照してください。

- 7** NCI セキュリティファイルを復元した場合は、復元完了後に NCI を再初期化するため、サーバを再起動します。

- 8** (状況によって実行) このサーバでロールフォワードログ機能を使うためには、改めて有効に切り替え、障害対策のための書き出し先も設定し直して、ロールフォワードログの環境設定を再作成する必要があります。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

この手順が必要となるのは、復元処理の過程で、ロールフォワードログに関する設定はデフォルトに戻るためです。つまり、ロールフォワードログ機能は無効となり、保存先もデフォルトの場所になるからです。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

ロールフォワードログの詳細については、[397 ページの「ロールフォワードログを使用する」](#) を参照してください。



## バックアップ / 復元の運用例

- ◆ 437 ページの「シナリオ : 単一サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合」
- ◆ 438 ページの「シナリオ : 複数サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合」
- ◆ 440 ページの「シナリオ : 複数サーバ構成のネットワークで、1 台のサーバが完全に使えなくなった場合」
- ◆ 441 ページの「シナリオ : 複数サーバ構成のネットワークで、数台のサーバが使えなくなった場合」
- ◆ 441 ページの「シナリオ : 複数サーバ構成のネットワークで、すべてのサーバが使えなくなった場合」

### シナリオ : 単一サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合

あるユーザは Stationery Supply 社で単一サーバ構成のネットワークを管理しています。サーバは 1 台しかないので、レプリカ機能で障害に備えるわけにはいきません。その場合、eDirectory 8.7.3 に組み込まれた新ツール Backup eMTool を使えば、eDirectory のバックアップ / 復元処理が容易になります。サーバ単位で処理を行う方式であり、しかも高速だからです。

これまで Windows NT Server で eDirectory 8.6.2 を稼働させていましたが、8.7.3 にアップグレードした結果、Backup eMTool を起動するバッチファイルを用意し、定期的に無人でバックアップを行えるようになりました。

毎週日曜の夜にフルバックアップを取り、さらに平日は毎晩、インクリメンタルバックアップを取るようになりました。また、eDirectory の無人バックアップのすぐ後に、ファイルシステムのフル / インクリメンタルバックアップを取るようになります。バックアップテープには、ファイルシステムデータのほか、eDirectory の最新バックアップも保存されることとなります。さらに、データ保存サービス会社と契約して、バックアップテープを社外に保管することにしました。

毎週月曜日の朝、バックアップログを調べて、正常にフルバックアップが取れていることを確認します。また、普段から時々、インクリメンタルバックアップの状況をログで調べています。

ロールフォワードログの機能は無効にしています。それは次の理由によります。

- ◆ サーバには独立した記憶デバイスがないので、ロールフォワードログを有効にしてもバックアップとしては役に立ちません。記憶デバイスが故障すれば、eDirectory ばかりでなくログも消えてしまうので、いずれにしても復元には使えません。
- ◆ ツリー構成が変わることはあまりありません。また、障害が発生しても、前の晩の状態にまで復元できれば充分だと考えています。停止直前の状態まで eDirectory を復元できなくても構いません。
- ◆ サーバはレプリカリングに属して他のサーバと連携しているわけではないため、ロールフォワードログがなくても、復元後の検証処理には成功するはずで

Stationery Supply 社では、人事異動などによりツリー構成を大きく変える場合は、その直前と直後に、手動でバックアップすることになっています。日曜日以外でも、必要に応じて臨時のバックアップを取ろうというわけです。これがロールフォワードログに代わる措置です。

必要になればいつでもバックアップ作業ができるよう、時々テストして確かめています。テスト用にもう 1 台サーバを購入する予算はないので、市内にあるサービス会社と契約し、必要なときだけサーバを使わせてもらえるようにしました。実機と似た構成のサーバにオペレーティングシステムをインストールし、eDirectory データベースの環境もできるだけ同じように構築します。このサーバに、実機から取ったバックアップファイルを使って復元し、想定どおりに復元されていることを確認するのです。

ある水曜日の朝、eDirectory が格納されたハードディスクの故障が見つかりました。そこで新しいハードディスクを手配したほか、日曜日を取ったフルバックアップファイル、月曜日および火曜日を取ったインクリメンタルバックアップファイルを用意しました。次に、ハードディスクを交換し、eDirectory をインストールしました。その上に、フル/インクリメンタルバックアップファイルから復元しました。水曜日の朝、故障が起こる前にツリー構成を変更しましたが、これは復元できませんでした。ロールフォワードログを残しておかなかったからです。しかし火曜日夜の状態まで復元できたことで満足しています。ロールフォワードログを取るための管理の手間を考えれば、許容できる範囲だと考えているからです。

## シナリオ：複数サーバ構成のネットワークで、eDirectory を格納しているハードディスクが故障した場合

あるユーザが Outdoor Recreation 社で eDirectory が稼動する 10 台のサーバを管理しています。毎週日曜の夜にフルバックアップを取り、さらに平日は毎晩、インクリメンタルバックアップを取っています。eDirectory のバックアップ後すぐに、ファイルシステムバックアップによりテープに保存しています。

サーバは 15 台ともレプリカリングに属しています。ロールフォワードログ機能は全サーバで有効です。また、その保存先は、eDirectory とは別の記憶デバイスに割り当てています。ディスクの空き容量やアクセス権は随時監視して、ロールフォワードログであふれてしまわないよう注意しています。使用済みのロールフォワードログは、時々テープにバックアップして削除し、空き容量を増やすようにしています。

もちろんロールフォワードログを有効にすると管理の手間が増えますが、それを上回る利点があると考えています。サーバがレプリカリングに属している場合、いつでも最新の状態をバックアップしておく必要があるからです。こうしておけば、障害が発生しても、他のサーバとの同期状態も含めて元どおりに復元できます。

さらに、テスト環境で定期的にバックアップファイルからの復元を試み、想定どおりに動作することを確認しています。

ある木曜日の午後 2 時、Inventory\_DB1 という Linux サーバの、eDirectory を格納しているハードディスクが故障しました。

そこでまず、最新のフルバックアップファイルとそれ以降のインクリメンタルバックアップファイルを集め、昨夜午前 1 時の状態にまでデータベースを復元します。ロールフォワードログは昨夜のバックアップ以降の変更を記録しています。そこで次に、午前 1 時以降のロールフォワードログを使って、故障が起こる直前の状態に戻さなければなりません。

次の手順で作業を進めることにしました。

1. まず、サーバのハードディスクを交換します。
2. 日曜の夜に取ったフルバックアップのテープを用意しました。

フルバックアップ用のバッチファイルは、ファイル /adminfiles/backup/backupfull.bk にバックアップするようになっています。

ファイルの容量制限を 200MB と設定していたので、次の 2 つのファイルがありました。

backupfull.bk.00001 (200MB)

backupfull.bk.00002 (32MB)

3. さらに、月曜、火曜、水曜に取ったインクリメンタルバックアップのテープも用意しました。

インクリメンタルバックアップ用のバッチファイルは、ファイル `/adminfiles/backup/backupincr.bk` にバックアップするようになっています。

毎日同じバッチファイルを使っているなので、ファイル名は常に同じです。しかし、テープからサーバにコピーする時に、ファイル名を変更しなければなりません。復元処理の際は、全ファイルを同じディレクトリに集めておく必要があるからです。

4. まず、ハードディスクを交換しました。

幸い Linux オペレーティングシステムを格納したハードディスクは故障しなかったので、Linux の再インストールは必要ありませんでした。

5. バックアップテープを使って、障害を受けたディスクパーティションを復元しました。
6. 次に eDirectory を再インストールし、仮のツリーを作りました。復元処理では、いったんこのツリー上にデータを復元してから実動ツリーに切り替えることとなります。
7. サーバ上に、バックアップファイルを集めておくためのディレクトリ `/adminfiles/restore` を作りました。
8. フルバックアップファイル (2 つに分割されているもの) を、このディレクトリにコピーしました。
9. さらに、月曜、火曜、水曜のインクリメンタルバックアップファイルもコピーしました。

どれも同じ `backupincr.bk` というファイル名なので、次のように改名します。

`backupincr.mon.bk`

`backupincr.tues.bk`

`backupincr.wed.bk`

**注:** インクリメンタルバックアップファイルは、必ずしもフルバックアップファイルと同じディレクトリにコピーしなくても構いません。ただし各インクリメンタルバックアップファイルはすべて同じディレクトリに置いておく必要があります。

10. 次に iManager を使って eDirectory を復元することにしました。
  - a. iManager を起動し、[eDirectory の保守] > [復元] の順にクリックします。
  - b. サーバにログインしました。コンテキストとしては、仮に作っておいたツリーを使います。
  - c. [復元ウィザード - ファイルの環境設定] 画面で次のように操作しました。

バックアップファイルの場所として、「`/adminfiles/restore`」を指定。

復元処理に関するログファイルの出力先として「`/adminfiles/restore/restore.log`」を指定。
  - d. [復元ウィザード - オプション] 画面で次のように操作しました。

[データベースを復元] チェックボックスをオン。

[ロールフォワードログの復元] チェックボックスをオン。

ロールフォワードログの保存先を入力。

これは普段ロールフォワードログを保存している場所です。eDirectory とは別のハードディスクなので、今までのロールフォワードログが残っているはずですが。

[セキュリティファイルの復元] チェックボックスをオン。

[検証後に復元されたデータベースをアクティブにします] チェックボックスをオン。

[復元の完了後にデータベースを開きます] チェックボックスをオン。

これは、復元後の検証に成功したら eDirectory をオープンするための指定です。

11. 復元処理を起動し、インクリメンタルバックアップファイル名を問い合わせるメッセージが表示されたので入力しました。
12. 復元後の検証処理にも成功し、自動的にデータベースがオープンされ、従来どおりのツリーで動作するようになりました。

ロールフォワードログはそのまま残っており、その場所を正しく指定したため、停止直前の状態に復元でき、検証も成功しました。

13. 復元後、ロールフォワードログに関する設定をやり直し、改めてフルバックアップを取っておきました。

ロールフォワードログの機能は、復元処理の過程で無効になってしまうため、ここで有効にする必要があります。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起る可能性があるためです。

サーバの稼動状況を調べ、正常に動作していることを確認しました。

## シナリオ：複数サーバ構成のネットワークで、1 台のサーバが完全に使えなくなった場合

ユーザは GK Designs 社で 15 台のサーバを管理しています。毎週土曜の夜にフルバックアップを取り、さらに毎晩、インクリメンタルバックアップを取っています。eDirectory のバックアップ後すぐに、ファイルシステムバックアップによりテープに保存しています。

サーバは 15 台ともレプリカリングに属しています。ロールフォワードログ機能は全サーバで有効です。

ある日、漏電による火事のため、ある支店のサーバが 1 台、完全に使えなくなりました。幸い、このサーバのパーティションは、ひとつを除いてすべて、他のサーバにレプリカが作成されていました。ロールフォワードログ機能は有効にしていたのですが、それも使えなくなりました。したがって、停止直前の状態にまで eDirectory データベースを復元することはできません。

しかしバックアップファイルがあるので、サーバの eDirectory 識別情報は再作成できます。ロールフォワードログは使えないので、他のサーバとの同期状態は元に戻せません (395 ページの「[遷移ベクトルと復元後の検証処理](#)」を参照)。検証処理には失敗するはずですが、復元処理が終わっても eDirectory データベースはオープンされません。

そこで、レプリカリングからいったんサーバを外し、DSRepair を使って、他のサーバを参照してレプリカを作る設定に変更しました。その結果、最新のレプリカを保持しているサーバからデータを参照し、最新の状態が作り出されます。具体的な手順は 432 ページの「[復元後の検証処理に失敗した場合の対処方法](#)」を参照してください。

レプリカが作られていなかったパーティションがひとつありました。これは、支店内にあるファックス/プリンタ複合機や大判カラープリンタなど、この支店内のネットワーク印刷オブジェクトを管理しているコンテナです。他のサーバにレプリカがないため、このパーティションは上述の手順では復元できません。このパーティションのオブジェクトは一から作り直さざるをえなかったため、将来に備え、これも他のサーバにレプリカを作っておくことにしました。

そこでロールフォワードログに関する設定をやり直し、改めてフルバックアップを取っておきました。ロールフォワードログの機能は、復元処理の過程で無効に戻ってしまうためです。

## シナリオ：複数サーバ構成のネットワークで、数台のサーバが使えなくなった場合

ユーザは3ヶ所に分けて設置された20台のサーバを管理しています。その1ヶ所で、水道管破裂による水漏れ事故のため、8台のうち5台が使えなくなってしまいました。

幸い、どのサーバについても、eDirectoryのバックアップを取っていました。しかし全サーバともレプリカリングに属しており、ロールフォワードログはなくなってしまったので、これを使わずに復元作業を進めなければなりません。最初にどのサーバから着手し、レプリカ間の不整合をどのように解消すればよいか、判断できませんでした。状況があまりに込み入っていたため、Novellの技術担当者に相談することにしました。

## シナリオ：複数サーバ構成のネットワークで、すべてのサーバが使えなくなった場合

ユーザとそのチームは、Human Resource Consulting社で1ヶ所に設置された50台のサーバを管理しています。

普段から障害対策のため、ツリーの各パーティションについて3つずつレプリカを作成しています。したがって、サーバが1台停止しても、そのパーティションにあるオブジェクトは他のサーバからアクセスできます。さらに、各サーバに障害があっても復元できるように、Backup eMToolで定期的にバックアップするほか、ロールフォワードログ機能を有効にし、バックアップテープは別の建物に保管しています。

災害に備えるため、チームでは2台のサーバをDSMASTERとして割り当てています。2台を割り当てているのは、ツリーが大きすぎて、1台では全パーティションのレプリカを保持できないからです。ツリーに属するどのパーティションも、いずれかのDSMASTERサーバにレプリカが作成されています。逆に、同じパーティションのレプリカが2台のDSMASTERサーバに作成されることはないようにして、重複を避けています。これが災害対策として重要な点です。

さらにチームでは、テスト環境で定期的にバックアップファイルからの復元を試み、想定どおりに動作することを確認しています。

ある日、台風で社屋が倒壊し、データセンタにあったサーバもすべて壊れてしまいました。

台風が去った後、チームではまず、パーティションすべてのレプリカを保持する、2台のDSMASTERサーバを復元する作業に取りかかりました。最新のフルバックアップファイルおよびそれ以降のインクリメンタルバックアップファイルを使用しましたが、ロールフォワードログは使いませんでした。サーバが壊れたときに、いっしょになくなってしまったからです。DSMASTERサーバを設定する際、この2台が同じレプリカを保持することはないようにしていました。そのため、ロールフォワードログを適用しなくても、復元後の検証処理は問題なく成功したのです。DSMASTERサーバが復元されたことにより、Human Resources Consulting社のツリーに属するオブジェクトはすべてアクセスできるようになりました。

DSMASTERが復元できれば、整合性を損なうことなくツリー全体を再作成できるので、その役割は非常に重要です。

このチームは、ロールフォワードログの機能も有効にしていました。障害が発生する直前の状態にまでサーバを復元でき、レプリカリングに属する他のサーバとの同期状態に不整合が生じないからです。サーバが復旧し、他のサーバと通信できるようになると、停止していた間に行われた更新情報を自動的に受け取り、同期を取ることができます。

しかし今回はそのロールフォワードログを使えません。その場合、レプリカリングに属するサーバのうち、最初に復元処理を実施したものしか正常に復元できないこととなります。それ以外のサーバは、他のサーバと同期状態が一致しないということで、復元後の検証処理に失敗してしまうのです (395 ページの「[遷移ベクトルと復元後の検証処理](#)」を参照)。検証処理に失敗すると、この eDirectory データベースはアクティブになりません。

しかしこのチームは、このような状況もきちんと考慮していました。2 台の DSMASTER サーバを設定し、重複して保持するパーティションがないようにして、これを復元作業の起点としたのです。重複がないので検証処理に失敗することはありません。これをマスタとして順次他のサーバにコピーしていけば、レプリカリング全体が復元できることとなります。

DSMASTER サーバの復元後、他のサーバを復元するために、いくつか必要な作業があります。このチームは次の手順で、他のサーバを順次復元していきました。

- ◆ DSMASTER サーバ上のレプリカを、マスタレプリカとして割り当てます。
- ◆ DSMASTER 以外のサーバをレプリカリングから外します。
- ◆ フル/インクリメンタルバックアップファイルを使って、DSMASTER 以外のサーバを復元します。

ロールフォワードログがないため、復元後の検証処理に失敗することは分かっています。したがって、データベースは復元されても、まだアクティブになっていません。

- ◆ 高度な復元機能を使って、データベースをアクティブにしました。ロックはまだ解除しません。
- ◆ DSRepair を使って、他のサーバを参照してレプリカを作る設定に変更しました。
- ◆ データベースのロックを解除しました。

この時点で、各サーバの識別情報は元どおりに戻っていますが、レプリカ情報を同期させようとはしません。他のサーバからレプリカのデータを受け取り、改めて構築できる状態になっています。

なお、NetWare サーバの場合、ファイルシステムの復元は eDirectory の復元後に行う必要があります。

- ◆ 各サーバは DSMASTER サーバからデータを受け取り、自動的にレプリカを復元してレプリカリングに復帰します。

各サーバにどのレプリカが置かれていたか、チームではきちんと把握していました。もっとも、最終バックアップ時点の状況は、バックアップファイルのヘッダを見れば確認できます。

- ◆ ロールフォワードログに関する設定をやり直し、改めてフルバックアップを取っておきました。ロールフォワードログの機能は、復元処理の過程で無効に戻ってしまうので、ここで有効にしたのです。改めてフルバックアップを取る必要があるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。

具体的な手順は [432 ページの「復元後の検証処理に失敗した場合の対処方法」](#) を参照してください。

かなりの作業量でしたが、ツリー自身は比較的早期に使用できるようになり、その時点でサーバ全体を復旧する目処も立っていました。

# 15

## Novell eDirectory のメンテナンス

Novell® eDirectory™ のパフォーマンスを最適にするには、定期的なヘルスチェック手順を実行し、必要に応じてハードウェアのアップグレードや交換を行ってディレクトリをメンテナンスする必要があります。

この章では、次のメンテナンスに関するトピックについて説明します。

### パフォーマンス

- ◆ 443 ページの「eDirectory のパフォーマンスの改善」
- ◆ 451 ページの「Linux、Solaris、AIX、および HP-UX システムでの eDirectory パフォーマンスの改善」

### ヘルスチェック

- ◆ 457 ページの「eDirectory の正常な動作の維持」
- ◆ 460 ページの「監視のためのリソース」

### ハードウェアの交換

- ◆ 460 ページの「ハードウェアのアップグレードやサーバの交換」

### eDirectory の回復

- ◆ 467 ページの「ハードウェア障害後の eDirectory の復元」

## eDirectory のパフォーマンスの改善

eDirectory のパフォーマンスに最も大きく影響するのは、キャッシュの設定です。NDS® の以前のバージョンでは、ブロックキャッシュ制限を指定することにより、eDirectory がキャッシュに使用するメモリ量を制御できました。デフォルトでは、8MB の RAM がキャッシュに使用されていました。

eDirectory 8.5 以降では、ブロックキャッシュ制限およびエン트리キャッシュ制限を指定できます。ブロックキャッシュは NDS の前のバージョンでも使用可能で、データベースの物理ブロックのみをキャッシュします。エン트리キャッシュは eDirectory 8.5 に導入された機能で、データベースの論理エントリをキャッシュします。エントリをキャッシュすることで、ブロックキャッシュからメモリ内にエントリをインスタンス化するために要する処理時間を削減できます。

この2つのキャッシュにはいくらか重複する部分がありますが、各キャッシュはそれぞれ異なる操作でのパフォーマンスを向上させるために設計されています。ブロックキャッシュは、更新操作で最も効果的です。エン트리キャッシュは、名前解決など、エントリを読み込んで eDirectory ツリーをブラウズする操作で最も効果的です。

ブロックキャッシュとエン트리キャッシュのいずれも、クエリのパフォーマンスを向上させます。ブロックキャッシュは、インデックス検索を高速化します。エン트리キャッシュは、インデックスから参照されるエントリの取得を高速化します。

eDirectory 8.7.3 のデフォルト値を次に示します。

- ◆ eDirectory をインストールするサーバにレプリカがない場合、デフォルトでは 16MB のハードメモリ制限が設定され、ブロックキャッシュ用に 8MB、エントリキャッシュ用に 8MB が使用されます。

詳細については、[445 ページの「ハードメモリ制限について」](#)を参照してください。

- ◆ サーバにレプリカがある場合、デフォルトでは使用可能メモリの 51% という動的調整制限が設定され、最小しきい値は 8MB、最大しきい値は 24MB になります。

詳細については、[444 ページの「動的調整制限について」](#)を参照してください。

## エントリキャッシュとブロックキャッシュでメモリを配分する

キャッシュ用に使用可能な全メモリが、エントリキャッシュとブロックキャッシュとで共有されます。デフォルトでは、均等に配分されます。NDS 8 の以前のバージョンで使用できたブロックキャッシュ量を維持するには、eDirectory の合計キャッシュサイズを 2 倍にする必要があります。LDIF インポートのパフォーマンス向上のためにキャッシュを使用する場合、合計キャッシュサイズを 2 倍にするか、デフォルトのキャッシュ設定を変更します。デフォルトのキャッシュ設定を変更するには、[445 ページの「動的調整制限およびハードメモリ制限を設定する」](#)を参照してください。

キャッシュ可能なブロック数およびエントリ数が多いほど、全体のパフォーマンスは向上します。大きなデータベースの場合には不可能ですが、データベース全体をエントリキャッシュおよびブロックキャッシュにキャッシュできれば理想的です。一般的に、ブロックキャッシュと DIB セットの比率ができるだけ 1:1 になるように設定します。エントリキャッシュについては、1:2 または 1:4 の比率になるようにします。最適なパフォーマンスを得るためには、これらの比率を上回るようにしてください。

## デフォルトのキャッシュ設定を使用する

eDirectory には、キャッシュメモリの使用量を制御する方法として、動的調整制限とハードメモリ制限の 2 つの方法があります。どちらの方法も使用できますが、この 2 つは互いに排他的であるため同時に両方を使用することはできません。最後に使用した方法により、常に以前の設定が置き換えられます。

### 動的調整制限について

動的調整制限では、eDirectory は他のプロセスによるメモリ消費量の増減に応じて、定期的にメモリ使用量を調整します。制限は、使用可能な物理メモリの割合として指定します。eDirectory はこの割合を使用し、一定の間隔で新しいメモリ制限を再計算します。新しいメモリ制限は、その時点で使用可能な物理メモリの割合として算出されます。

割合とともに、しきい値として最大値および最小値も設定できます。しきい値はバイト数で指定し、eDirectory がその値に調整します。しきい値には、使用するバイト数、または使用可能量として残すバイト数のいずれかが指定できます。デフォルトの最小しきい値は 16MB です。デフォルトの最大しきい値は 4GB です。

最小しきい値と最大しきい値による制限が矛盾する場合は、最小しきい値が優先されます。たとえば、次のような設定を指定したとします。

最小しきい値	8MB
使用する利用可能な物理メモリの割合	75
最大しきい値	10MB を使用可能量として維持



eDirectory がキャッシュ制限を調整するときに、使用可能な物理メモリが 16MB あるとします。eDirectory は、新しい制限値を 12MB として算出します。eDirectory は、算出した新しい制限値が最小しきい値と最大しきい値の範囲内にあるかをチェックします。この例では、最大しきい値の設定により 10MB を使用可能量として残しておく必要があるため、eDirectory は制限値を 6MB に設定します。しかし、最小しきい値は 8MB であるため、eDirectory は最終的に制限値を 8MB に設定します。

動的調整制限では、間隔の長さも指定します。デフォルトの間隔は 15 秒です。間隔が短いほど、より最新の状態に基づいてメモリが使用されます。ただし、割合の再計算によりメモリの割り当てと解放が発生するため、必ずしも間隔が短いほど良いわけではありません。

## ハードメモリ制限について

ハードメモリ制限は、eDirectory の以前のバージョンで、メモリ消費を制御するために使用されている方法です。次のいずれかの方法でハードメモリ制限を設定します。

- ◆ 固定バイト数
- ◆ 物理メモリの割合  
その期間の物理メモリの割合が、固定バイト数になります。
- ◆ 使用可能な物理メモリの割合  
その期間の使用可能な物理メモリの割合が、固定バイト数になります。

## キャッシュをクリーンアップする


NDS 8 はトランザクションの整合性を維持するために、キャッシュ内に複数のバージョンのブロックとエントリを作成します。NDS 8 の以前のバージョンでは、これらのブロックやエントリは不要になっても削除されませんでした。eDirectory 8.7.3 では、バックグラウンドプロセスにより定期的にキャッシュがブラウズされ、古いバージョンが消去されます。これにより、キャッシュメモリの消費量が最小限に抑えられます。デフォルトのブラウズ間隔は 15 秒です。

## 動的調整制限およびハードメモリ制限を設定する

動的調整制限およびハードメモリ制限は、次のいずれかの方法で設定できます。

- ◆ [445 ページの「Novell iMonitor を使用する」](#)
- ◆ [447 ページの「\\_ndsdb.ini ファイルの使用」](#)

### Novell iMonitor を使用する

- 1 [エージェント環境設定]  をクリックします。
- 2 [データベースキャッシュ] をクリックし、次の情報を確認します。

データベースキャッシュ情報	説明
最大サイズ	指定したキャッシュが拡張できる最大サイズ (KB) です。
現在のサイズ	指定したキャッシュの現在のサイズ (KB) です。
キャッシュされたアイテム	指定したキャッシュ内のアイテム数です。

データベースキャッシュ情報	説明
キャッシュされた古いバージョン	指定したキャッシュ内の古いバージョンの数です。古いバージョンのキャッシュアイテムは、データベースの読み込みトランザクションの整合性を維持するために保持されます。つまり、あるスレッドが読み込みトランザクションを実行しており、別のスレッドが書き込みトランザクションを実行している場合、書き込みによって変更されるブロックの古いバージョンが読み込み操作のために保持されます。これは、読み込みのトランザクションを実行している間に変更処理が発生したとしても、読み込みの表示結果に整合性があるようにするためです。
古いバージョンのサイズ	キャッシュされる古いバージョンのアイテムサイズ(KB)です。
ヒット	指定したキャッシュ内のアイテムに正常にアクセスできた回数です。
ヒット表示	指定したキャッシュ内で、アイテムに正常にアクセスする前に参照されたアイテム数です。ヒット表示とヒットの比率が、キャッシュ検索効率の目安となります。通常、この比率はほぼ 1:1 になるようにします。
失敗	アイテムが指定したキャッシュ内に見つからず、低いレベルのキャッシュまたはディスクから取得されなければならなかった回数です。
失敗表示	必要なアイテムが指定したキャッシュ内にないと判断されるまでに、キャッシュ内で参照されたアイテムの数です。失敗表示と失敗の比率が、キャッシュ検索効率の目安となります。通常、この比率はほぼ 1:1 になるようにします。

### 3 次のオプションから選択します。

オプション	説明
ダイナミック調整	eDirectory データベースのキャッシュに使用するシステムメモリの大きさを、必要と判断される大きさと、次に示すパラメータに基づいて動的に調整します。
キャッシュ調整パーセンテージ	レコードキャッシュおよびブロックキャッシュに使用可能なメモリの割合です。
キャッシュサイズの制約条件	ダイナミック調整で、指定した制約条件に従います。つまり、キャッシュのメモリの大きさを指定した大きさより小さくせず、使用可能なメモリの合計から指定した大きさを引いた値より大きくしません。
ハードメモリ制限	キャッシュに使用するシステムメモリの正確な大きさです。
キャッシュ最大サイズ	レコードキャッシュとブロックキャッシュを合わせたサイズ (KB) です。
ブロックキャッシュパーセンテージ	ブロックキャッシュに割り当てられる使用可能なシステムメモリの割合です。残りの割合が、レコードキャッシュに割り当てられます。
キャッシュ調整間隔	この間隔は、ダイナミック調整を設定している場合のみ使用されます。調整間隔は、指定したパーセンテージと制約条件に基づき、どれくらいの頻度でキャッシュサイズを調整するかを制御します。

オプション	説明
キャッシュクリーンアップ間隔	どれくらいの頻度で古い未使用のバージョンをキャッシュから削除するかを制御します。
永続的なキャッシュ設定	このオプションを選択すると、iMonitor を使用して送信した変更は、前に保存した設定やデフォルトのシステム設定に上書きされ、永続的な変更になります。

4 [送信] をクリックします。

## \_ndsd.db.ini ファイルの使用

1 \_ndsd.db.ini をテキストエディタで開きます。

NetWare® では、このファイルは `sys:\netware` にあります。通常、Windows NT および Windows 2000 では、このファイルは `\Novell\NDS\DIBfiles` にあります。

2 ファイルに該当する構文を追加します。

コマンド	変数の説明	定義
<code>cache=cache_bytes</code>	使用する固定バイト数	ハードメモリ制限を設定します。  たとえば、8MB のハードメモリ制限を設定するには、次のように入力します。  <code>cache=8000000</code>
<code>cache=cache_options</code>	複数のオプションをコンマで区切り、任意の順序で指定できます。  <ul style="list-style-type: none"> <li>◆ DYN 動的調整制限を設定します。</li> <li>◆ HARD ハードメモリ制限を設定します。</li> <li>◆ <code>%:percentage</code> 使用する利用可能なメモリまたは物理メモリの割合を設定します。</li> <li>◆ AVAIL または TOTAL 使用可能な物理メモリまたは合計物理メモリの割合を指定します。ハードメモリ制限においてのみ、使用できます。</li> <li>◆ <code>MIN:number_of_bytes</code> 最小バイト数。</li> <li>◆ <code>MAX:number_of_bytes</code> 最大バイト数。</li> <li>◆ <code>LEAVE:number_of_bytes</code> 使用可能量として残す最小バイト数。</li> </ul>	ハードメモリ制限または動的調整制限を設定します。  たとえば、動的調整制限で使用可能メモリを 75% に設定し、最小バイト数を 16GB に設定するには、次のように入力します。  <code>cache=DYN, %:75,MIN:16000000</code>  また、ハードメモリ制限で合計物理メモリの 75% を設定し、最小バイト数を 16MB に設定するには、次のように入力します。  <code>cache=HARD,%:75,MIN: 16000000</code>

- 3 (オプション) 動的調整制限の間隔を指定するには、次の行を追加します。

```
cacheadjustinterval=number_of_seconds
```

- 4 (オプション) エントリとブロックの古いバージョンをクリーンアップする間隔を指定するには、次の行を追加します。

```
cachecleanupinterval=number_of_seconds
```

- 5 (オプション) ブロックキャッシュとエントリキャッシュでのメモリ配分の割合を変更するには、次の行を追加します。

```
blockcachepcent=percent
```

変数 *percent* は、0 ~ 100 の範囲で指定する必要があります。ここで指定する割合は、ブロックキャッシュに使用するキャッシュメモリのパーセンテージです。残りの割合がエントリキャッシュ用に使用されます。割合を 0 に設定することはお勧めできません。

- 6 変更は、eDirectory サーバの再起動後に有効になります。

## DSTrace を使用して制限を設定する

eDirectory for NetWare を使用している場合、DSTrace で動的調整制限やハードメモリ制限を設定できます。変更を有効にするために、サーバを再起動する必要はありません。

- 1 (オプション) 固定バイト数のハードメモリ制限を設定するには、サーバコンソールで次のように入力します。

```
SET DSTTRACE=!MBamount_of_RAM_to_use_in_bytes
```

たとえば、8MB のハードメモリ制限を設定するには、次のように入力します。

```
SET DSTTRACE=!MB8388608
```

- 2 (オプション) パーセンテージによって算出されるハードメモリ制限を設定するには、サーバコンソールで次のように入力します。オプションは、指定するものだけを入力します。

```
SET DSTTRACE=!MHARD,AVAIL OR  
TOTAL, %:percent, MIN:number_of_bytes, MAX:number_of_  
bytes, LEAVE:number_of_bytes_to_leave, NOSAVE
```

たとえば、ハードメモリ制限で合計物理メモリの 75% を設定し、最小バイト数を 16MB に設定して、これらのオプションをスタートアップファイルに保存しないように指定するには、次のように入力します。

```
SET DSTTRACE=!MHARD, %:75, MIN:16777216, NOSAVE
```

- 3 (オプション) 動的調整制限を設定するには、サーバコンソールで次のように入力します。

```
SET DSTTRACE=!MDYN, %:percent, MIN:number_of_bytes, MAX:  
number_of_bytes, LEAVE:number_of_bytes_to_leave,  
NOSAVE
```

たとえば、動的調整制限で使用可能メモリの 75% を設定し、最小値を 8GB に設定するには、次のように入力します。

```
SET DSTTRACE=!MDYN, %:75, MIN:8388608
```

## LDAP for eDirectory をチューニングする

基本的な LDAP サーバのハードウェアとソフトウェア設定、チューニングパラメータ、およびディレクトリ編成のヒントについての詳細は、『[How to Configure and Optimize eDirectory LDAP Servers\(LDAP サーバを環境設定および最適化する方法\)](http://developer.novell.com/research/appnotes/2000/septembe/04/a000904.htm) (<http://developer.novell.com/research/appnotes/2000/septembe/04/a000904.htm>)』を参照してください。

### メモリを管理する

eDirectory は、データベースキャッシュとディレクトリのためにメモリを使用します。これらはメモリプールとしてそれぞれに割り当てられます。ディレクトリエンジンは、オペレーティングシステムで利用できるメモリプールから必要なメモリを取り込んで使用します。データベースは、次に詳しく説明するパラメータにより定義されたキャッシュプールを使用します。一般に、eDirectory に割り当てるデータベースキャッシュの量が多いほどパフォーマンスはよくなります。ただし、eDirectory はバッファにシステムメモリを使用するため、クライアントがクエリを実行し、そのクエリが大量のデータを戻す場合には、ディレクトリで大量のクエリ応答を行えるように、データベースキャッシュのサイズを小さくして、システムメモリを確保しなければならない場合があります。

データベースエンジンは、データベースキャッシュを使用して、最近アクセスしたブロックを保持します。このキャッシュの初期定義サイズは固定値で 16MB です。このキャッシュのサイズは、出荷バージョンの eDirectory でコマンドラインから変更できます。次の例は、eDirectory データベースキャッシュのサイズを 80MB に設定するコマンドです。

```
set dstrace=!mb 80000000
```

あるいは、NetWare サーバ上の SYS:\\_netware ディレクトリまたは Windows や UNIX の環境における eDirectory データベースファイルが格納されたディレクトリ (Windows では通常、`install directory\novell\nds\dbfiles`、UNIX では通常、`\var\nds\dib`) に `_ndsdb.ini` という名前のファイルを定義することもできます。このテキストファイルに、次のような行を設定するだけです。

```
cache=80000000
```

等号 (=) の前後にスペースは入れないでください。

eDirectory 8.7.3 のキャッシュは、以前のバージョンと同様に、ハードメモリ制限値に基づいて初期化できます。さらに、上限と下限はハード値で設定できるほか、使用可能なメモリの割合でも設定できます。動的割り当て用の制御パラメータを使用すれば、キャッシュサイズを使用状態に応じて増減させることができます。適切なパラメータを設定すれば、データベースキャッシュは他のシステムリソースからの要求に応じて、そのサイズを増減させます。

`_ndsdb.ini` ファイルを変更することにより、データベースメモリの使用を手動で制御できます。INI ファイルコマンドの形式は次のとおりです。

```
cache=cacheBytes # Set a hard memory limit
```

他の形式を次の表に示します。

コマンド	説明
<code>cache=cache_options</code>	ハードメモリ制限または動的調整制限を設定します。複数のキャッシュオプションをコンマで区切り、任意の順序で指定できます。これらはいずれも必須ではありません。キャッシュオプションを次に示します。
DYN または HARD	動的制限またはハードメモリ制限。
AVAIL または TOTAL	ハードメモリ制限が選択された場合にのみ有効です。動的調整制限の場合は、これらのオプションは省略します。
<code>%:percentage</code>	使用可能な物理メモリまたは合計物理メモリの割合。
<code>MIN:bytes</code>	最小バイト数。
<code>MAX:bytes</code>	最大バイト数。
<code>LEAVE:bytes</code>	OS 用として残す最小バイト数。
<code>blockcachepersent=percentage</code>	キャッシュをブロックキャッシュおよびレコードキャッシュに分離します。

ハードメモリ制限が指定された状態で、メモリの割合としてデータベースキャッシュを定義する場合、管理者は、合計メモリの割合または使用可能なメモリの割合のいずれかを選択できます。動的制限の場合には、必ず利用可能なメモリが基準になります。次に示すコマンド例はすべて、`_ndsdb.ini` ファイルで使用できます。

次の例では、動的制限を使用可能なメモリの 75%、最小バイト数を 16MB、OS 用に残すバイト数を 32MB と設定しています。

```
cache=DYN,%:75,MIN:16000000, LEAVE 32000000
```

次の例では、ハードメモリ制限を合計物理メモリの 75%、最小バイト数を 18MB、最大バイト数を 512MB と設定しています。

```
cache=HARD, TOTAL,%:75,MIN:18000000, MAX 512000000
```

次の例は古いスタイルのコマンドで、ハードメモリ制限を 8MB に設定しています。

```
cache=8000000
```

データベースキャッシュは、ブロックキャッシュとレコードキャッシュに配分されます。ブロックキャッシュには、ディスクに格納されたデータのミラーとして、データブロックとインデックスブロックが保持されます。レコードキャッシュには、メモリに展開された形でのディレクトリオブジェクトと属性が保持されます。ディレクトリへの追加や更新では、ブロックキャッシュの設定が使用されます。大半の処理が読み込みの場合には、レコードキャッシュが使用されます。適切なサイズのキャッシュを割り当てないで多数の順次更新処理を実行すると、両方のキャッシュにスラッシュ状態を招くことがあります。指定によって変更しない場合、キャッシュの 50% がブロックキャッシュに、50% がレコードキャッシュに割り当てられます。`blockcachepersent` オプションを `_ndsdb.ini` ファイルに追加することにより、データブロックとインデックスブロックに割り当てるキャッシュの比率を指定できます (デフォルトは 50% です)。残りのキャッシュ領域はエントリに使用されます。

たとえば、60% をブロックキャッシュに、40% をレコードキャッシュにするには、次のように入力します。

```
blockcachepersent=60
```

どちらかのキャッシュを 100% に設定して、他方のキャッシュが使用できないようにすることは避けてください。通常、一方のキャッシュに 75% 以上は設定しないようにしてください。

データベースキャッシュの設定は Novell iMonitor を使用して制御することもできます。

キャッシュサイズは使用できるメモリの量により動的に変動しますが、カスタム環境用に DSTRACE コマンドを使用することは可能です。

## Linux、Solaris、AIX、および HP-UX システムでの eDirectory パフォーマンスの改善

次のセクションでは、UNIX システム上で eDirectory のパフォーマンスを改善する方法について説明します。

- ◆ [451 ページの「eDirectory サーバを微調整する」](#)
- ◆ [452 ページの「eDirectory のキャッシュを最適化する」](#)
- ◆ [455 ページの「データのバルクロードを最適化する」](#)
- ◆ [456 ページの「Novell eDirectory 用に Solaris OS をチューニングする」](#)

### eDirectory サーバを微調整する

Linux および Solaris 上の Novell eDirectory は、動的に調整されるスレッドプールを使用して、クライアントの要求を処理します。スレッドプールは自動的に調整され、多くの場合は最適なパフォーマンスが提供されます。ただし、次のパラメータを `/etc/nds.conf` ファイルに設定することによって、サーバへの負荷が急激に高くなった場合にスレッドの起動により発生する遅延を回避できます。

パラメータ	説明および推奨される設定
<code>n4u.server.idle-threads</code>	最小スレッド数 (アクティビティとは無関係)。  このパラメータ値は、クライアントアクティビティが通常の場合に、新しいスレッドの生成に必要とされる時間を最小にするように、平均クライアント負荷に基づいて決める必要があります。
<code>n4u.server.max-threads</code>	最大スレッド数。  このパラメータ値は、同時にサービスを提供する必要があるクライアントの最大数に基づいて決める必要があり、次のスレッドを考慮することが推奨されます。 <ul style="list-style-type: none"><li>◆ eDirectory は、最低でも 16 のスレッドを必要とします。</li><li>◆ 255 の LDAP 接続それぞれに、1 つのスレッドが必要です (Monitor Thread: 監視スレッド)。</li><li>◆ 現在のクライアントのそれぞれに、1 つのスレッドが必要です (Worker Thread: 動作スレッド)。</li></ul>
<code>n4u.server.start-threads</code>	eDirectory の起動時に起動するスレッド数  このパラメータ値は、クライアントアクティビティが通常の場合に、新しいスレッドの生成に必要とされる時間を最小にするように、平均クライアント負荷に基づいて決める必要があります。

## eDirectory のキャッシュを最適化する

Novell eDirectory では永続キャッシュを使用しているため、サーバに対して行われた変更はベクトルに保持されます。変更の途中でサーバがクラッシュした場合、サーバが稼働状態に戻ると、eDirectory はより高速でロードして数秒間で変更を同期させます。Novell eDirectory は、システムエラーが発生した場合、ロールバックモデルとログファイルを使用してトランザクションをロールフォワードします。

eDirectory のキャッシュは最初 16MB に設定され、その 50% がブロックキャッシュに、残りの 50% がレコードキャッシュに割り当てられます。15 分後、eDirectory はキャッシュのしきい値を変更し、キャッシュに使用可能な空きメモリの 51% まで初期化できるようにしますが、少なくとも 24MB は OS 用に残します。このアルゴリズムは、使用可能な空きメモリ量を判別できるようにするコールをホスト OS がサポートしている場合のみ使用されます。

eDirectory キャッシュの最適化は、次の方法で行えます。

- ◆ 452 ページの「UNIX システムにおける固定量の RAM 使用」
- ◆ 454 ページの「キャッシュパラメータの設定」
- ◆ 455 ページの「データのバルクロードを最適化する」
- ◆ 455 ページの「LBURP トランザクションサイズを最適化する」

### UNIX システムにおける固定量の RAM 使用

上記のアルゴリズムは Windows および NetWare では有効ですが、UNIX システムでは機能しません。UNIX システムでは、OS によって報告される使用可能な空きメモリは、他のオペレーティングシステムよりも少なくなります。これは、UNIX OS がファイルシステムブロック、頻繁に実行するプログラム、ライブラリなどの内部キャッシング用に空きメモリを使用するためです。このメモリ割り当てに加え、通常、UNIX 上のライブラリは解放されたメモリを OS に戻しません。

このような理由で、キャッシュには固定量の RAM を割り当てることをお勧めします。

UNIX システムで固定量の RAM を割り当てるには、次のいずれかの操作を行います。

- ◆ 452 ページの「手動で .ini ファイルを作成する」
- ◆ 453 ページの「Novell iMonitor を使用する」


#### 手動で .ini ファイルを作成する

- 1 eDirectory データベースファイル (DIB セット) が格納されたディレクトリ (通常、/var/nds/dib) と同じディレクトリに、\_ndsdb.ini という名前のファイルを作成します。
- 2 次に示すパラメータを、\_ndsdb.ini ファイルに追加します。

パラメータ	説明
blockcachepersent=50	データベースブロックのキャッシングに割り当てるキャッシュの割合を設定します。
cacheadjustinterval=15	eDirectory が空きメモリの利用状況を評価し、全体のキャッシュサイズを調整する時間 (秒) を設定します。
cachecleanupinterval=15	eDirectory がダーティキャッシュブロックをディスクに書き込む時間 (秒) を設定します。
cache=16777216	ハードメモリ制限 (バイト数) を設定します。



## Novell iMonitor を使用する

- 1 [エージェント環境設定]  をクリックします。
- 2 [データベースキャッシュ] をクリックし、次の情報を確認します。

データベースキャッシュ情報	説明
最大サイズ	指定したキャッシュが拡張できる最大サイズ (KB) です。
現在のサイズ	指定したキャッシュの現在のサイズ (KB) です。
キャッシュされたアイテム	指定したキャッシュ内のアイテム数です。
キャッシュされた古いバージョン	指定したキャッシュ内の古いバージョンの数です。古いバージョンのキャッシュアイテムは、データベースの読み込みトランザクションの整合性を維持するために保持されます。つまり、あるスレッドが読み込みトランザクションを実行しており、別のスレッドが書き込みトランザクションを実行している場合、書き込みによって変更されるブロックの古いバージョンが読み込み操作のために保持されます。これは、読み込みのトランザクションを実行している間に変更処理が発生したとしても、読み込みの表示結果に整合性があるようにするためです。
古いバージョンのサイズ	キャッシュされる古いバージョンのアイテムサイズ (KB) です。
ヒット	指定したキャッシュ内のアイテムに正常にアクセスできた回数です。
ヒット表示	指定したキャッシュ内で、アイテムに正常にアクセスする前に参照されたアイテム数です。ヒット表示とヒットの比率が、キャッシュ検索効率の目安となります。通常、この比率はほぼ 1:1 になるようにします。
失敗	アイテムが指定したキャッシュ内に見つからず、低いレベルのキャッシュまたはディスクから取得されなければならなかった回数です。
失敗表示	必要なアイテムが指定したキャッシュ内にないと判断されるまでに、キャッシュ内で参照されたアイテムの数です。失敗表示と失敗の比率が、キャッシュ検索効率の目安となります。通常、この比率はほぼ 1:1 になるようにします。

- 3 次のオプションから選択します。

オプション	説明
ダイナミック調整	eDirectory データベースのキャッシュに使用するシステムメモリの大きさを、必要と判断される大きさと、次に示すパラメータに基づいて動的に調整します。
キャッシュ調整パーセンテージ	レコードキャッシュおよびブロックキャッシュに使用可能なメモリの割合です。
キャッシュサイズの制約条件	ダイナミック調整で、指定した制約条件に従います。つまり、キャッシュのメモリの大きさを指定した大きさより小さくせず、使用可能なメモリの合計から指定した大きさを引いた値より大きくしません。

オプション	説明
ハードメモリ制限	キャッシュに使用するシステムメモリの正確な大きさです。
キャッシュ最大サイズ	レコードキャッシュとブロックキャッシュを合わせたサイズ (KB) です。
ブロックキャッシュパーセンテージ	ブロックキャッシュに割り当てられる使用可能なシステムメモリの割合です。残りの割合が、レコードキャッシュに割り当てられます。
キャッシュ調整間隔	この間隔は、ダイナミック調整を設定している場合のみ使用されます。調整間隔は、指定したパーセンテージと制約条件に基づき、どれくらいの頻度でキャッシュサイズを調整するかを制御します。
キャッシュクリーンアップ間隔	どれくらいの頻度で古い未使用のバージョンをキャッシュから削除するかを制御します。
永続的なキャッシュ設定	このオプションを選択すると、iMonitor を使用して送信した変更は、前に保存した設定やデフォルトのシステム設定に上書きされ、永続的な変更になります。

#### 4 [送信] をクリックします。

### キャッシュパラメータの設定

デフォルトでは、eDirectory は動的キャッシュを使用します。eDirectory のキャッシュサイズの増加に使用できる十分な RAM がある場合は、eDirectory キャッシュへの RAM の割り当てを増加することによって、大容量データベースを使用する場合の eDirectory のパフォーマンスを大幅に向上できます。

次の表に示すパラメータを調整すれば、eDirectory のパフォーマンスを向上できます。

eDirectory キャッシュパラメータ	説明
blockcachepercent= <i>value</i>	データベースブロックのキャッシングに割り当てるキャッシュの割合を設定します。デフォルトは 50 です。
cachecleanupinterval= <i>value</i>	eDirectory がダーティキャッシュブロックをディスクに書き込む時間 (秒) を設定します。デフォルトは 15 です。
cacheadjustinterval= <i>value</i>	eDirectory が空きメモリの利用状況を評価し、全体のキャッシュサイズを調整する時間 (秒) を設定します。デフォルトは 15 です。
cache= <i>value</i>	eDirectory がキャッシングに使用できるメモリのハード制限 (バイト数) を設定します。
cache=leave: <i>value</i>	使用可能量として残しておく最小バイト数を指定します。
min: <i>value</i>	最小キャッシュサイズをバイト数で指定します。
max: <i>value</i>	最大キャッシュサイズをバイト数で指定します。

このアルゴリズムに従うと、Novell eDirectory のデフォルト設定は次のようになります。

```
cache=dyn,%:51,min:16777216,max:0,leave:0
```

この設定は、次のことを示しています。

- ◆ 最小キャッシュサイズは 16MB である。
- ◆ 最大キャッシュサイズの限度はない。
- ◆ 動的なキャッシュ調整が行われ、使用可能なメモリの 51% まで使用される。
- ◆ OS 用に 24MB が残される。

eDirectory は、すべてのアプリケーションが開始され、システムが安定するように、16MB のハードメモリ制限で動作します。

eDirectory が合計メモリの割合を使用するように設定することもできます。これを行うには、次のようにキャッシュを指定します。

```
cache=hard,total,%:percentage_of_total_memory_in_bytes
```

## データのバルクロードを最適化する

インポート/エクスポート変換(ICE)ユーティリティを使用するバルクロードのパフォーマンスは、アイテムの数によって影響されることがあります。パフォーマンス低下の最も一般的な原因には、ディスク I/O の管理不良や Novell eDirectory キャッシュのメモリ割り当て不足があります。

eDirectory が実質的な唯一のアプリケーションである場合、eDirectory キャッシュを合計メモリの 80% まで設定できます。割り当てられたキャッシュは、最終的にはすべてが使用されます。高揮発性のデータを処理している場合、eDirectory のパフォーマンスはキャッシュ量を増やすことで向上します。

**重要：**サーバが eDirectory 以外のアプリケーションやサービスのホストサーバとなっている場合は、合計メモリの 40% を超えるキャッシュメモリサイズを設定しないでください。テスト済みの最小キャッシュサイズは 0 で、最大キャッシュサイズは 3GB です。適切なキャッシュサイズは、同じサーバで実行される他の処理が必要とするメモリ量、および必要とするディスクキャッシュ量を基に決める必要があります。色々なキャッシュサイズを試してみて、最適なキャッシュサイズを決めてください。

バルクロードのパフォーマンスを最適にするには、eDirectory キャッシュの割り当てで、ブロックキャッシュにより高い割り当て率を設定します。ブロックキャッシュに 80% の値を設定することをお勧めします。この設定は、処理が完了した後でリセットできます。

iMonitor を使用すれば、最も迅速に blockcachepcentage パラメータを変更できます。この作業を行う方法については、[453 ページの「Novell iMonitor を使用する」](#)を参照してください。

## LBURP トランザクションサイズを最適化する

LBURP トランザクションサイズによって、1 つのトランザクションにおいて ICE から LDAP サーバに送信されるレコード数が設定されます。十分なメモリがあり、この値を大きくしても I/O 競合が発生しない場合、この値を大きくすることでバルクロードのパフォーマンスを向上できます。

デフォルトの LBURP トランザクションサイズは 25 です。この値は LDIF ファイルが少ない (操作数が 100,000 より少ない) 場合には適切ですが、レコード数が多い場合には不適切です。LBURP トランザクションサイズは、1 ~ 1000 の範囲で設定できます。

## トランザクションサイズの変更

トランザクションサイズを変更するには、`/etc/nds.conf` ファイルで `n4u.ldap.lburp.transize` パラメータの値を変更します。

理想的なシナリオでは、トランザクションサイズが大きいほど、パフォーマンスはより高くなります。ただし、次の理由のため、トランザクションサイズには必要以上に大きな値を設定しないようにします。

- ◆ トランザクションサイズが大きいほど、サーバはトランザクションを処理するためにより多くのメモリを割り当てる必要があります。システムが少ないメモリで稼働している場合、スワッピングのために処理が遅くなることがあります。
- ◆ LDIFファイルにエラーがないこと、およびeDirectoryにすでに存在するエントリがコメント化されていることを確認します。トランザクションに1つでもエラーがあると(追加しようとしたオブジェクトがすでにディレクトリに存在するという場合も含めて)、eDirectoryはLBURPトランザクション設定を無視し、操作ごとにコミットを実行してデータの整合性を確認します。

詳細については、[506 ページの「LDIF ファイルのデバッグ」](#)を参照してください。

- ◆ LBURPの最適化は、リーフオブジェクトに対してのみ有効です。トランザクションにコンテナオブジェクトとその従属オブジェクトが含まれている場合、eDirectoryはこれをエラーと見なします。これを回避するには、最初に別のLDIFファイルからコンテナオブジェクトをロードするか、または前方参照の使用を有効にします。

詳細については、[506 ページの「前方参照を有効化する」](#)を参照してください。

## Novell eDirectory 用に Solaris OS をチューニングする

次のセクションでは、Solaris のカーネル、ネットワーク、およびファイルシステムのチューニング方法について説明します。

**重要** : 最初に、推奨されているパッチを Solaris OS に適用済みであることを確認します。詳細については、『*Novell eDirectory 8.7.3 インストールガイド*』の「[Novell eDirectory の Solaris へのインストールまたはアップグレード](#)」を参照してください。

- ◆ [456 ページの「Solaris カーネルをチューニングする」](#)
- ◆ [457 ページの「Solaris ネットワークをチューニングする」](#)
- ◆ [457 ページの「Solaris ファイルシステムを微調整する」](#)

### Solaris カーネルをチューニングする

Solaris 上の eDirectory パフォーマンスを最適化するには、`/etc/system` ファイルで次のカーネル変数を設定します。

パラメータ	説明
<code>set maxphys=1048576</code>	1 回の SCSI 転送で転送できる最大バイト数です。
<code>set md_maxphys=1048576</code>	DiskSuite、 <code>vol_maxio</code> 、または <code>VxVM</code> を使用している場合の、1 回の SCSI 転送で転送できる最大バイト数です。
<code>set ufs:ufs_LW=1/128_of_available_memory</code>	1 つのファイルに対する未処理バイト数の境界値です。これを下回ると、他の処理が休眠状態となっている原因の条件変数が切り替えられます。
<code>set ufs:ufs_HW=1/64_of_available_memory</code>	1 つのファイルの書き込み失敗境界値に対する未処理バイト数です。

パラメータ	説明
ctcp:tcp_conn_hash_size=8192	カーネルのデータ構造を検索するために割り当てられた、TCP 接続に関連する接続ハッシュエントリの数です。(この数は、LDAP クライアントの数に応じて 262144 まで大きくできます)。

## Solaris ネットワークをチューニングする

LDAP の検索パフォーマンスは、Solaris の `ndd` コマンドを使用して改善できます。次に示すコマンド構文を使用すれば、ネットワークの操作や動作に影響する、チューニング可能なパラメータを分析し、変更できます。

```
ndd -set /dev/tcp variable_name variable_value
```

変数の推奨値を次の表に示します。

パラメータ	説明
tcp_conn_req_max_q: 1024	「q」はキューを表します。これは完成されたソケットで、ペンを保持し、アプリケーションが受諾を発行するまでソケットは継続します。
tcp_time_wait_interval: 60000	待機間隔 (ここでは小さな値) を設定します。
tcp_xmit_hiwat: 64000	TCP の最大および最小送信ウィンドウサイズを調整します。
tcp_xmit_lowat: 64000	
tcp_slow_start_initial: 2	最初の転送パケット数を 1 から 2 に調整します。

## Solaris ファイルシステムを微調整する

Solaris ファイルシステムが適切にチューニングされると、Solaris 上の Novell eDirectory のパフォーマンスが改善されます。特に、ディレクトリヘータをバルクロードする場合のパフォーマンスが非常に向上します。eDirectory のファイルシステムチューニングは、データベースのチューニングと類似しています。Solaris ファイルシステムの詳細については、Sunworld\* の Web サイト (<http://www.sunworld.com/sunworldonline>) を参照してください。

## eDirectory の正常な動作の維持

ディレクトリサービスの動作を正常に維持することは、あらゆる組織にとって重要です。Novell iMonitor を使用して定期的にヘルスチェックを行うことで、ディレクトリは適切に機能し、アップグレードやトラブルシューティングを容易に実行できます。

## ヘルスチェックを実行する時期

一般に、ネットワークを頻繁に変更しない場合 (サーバとパーティションの追加は 2 ~ 3ヶ月おきで、頻繁に行われるのは単純な変更だけの場合)、ヘルスチェックは月に 1 度実行します。

ネットワークの変更が頻繁に発生する場合 (パーティションやサーバが毎週追加される、あるいは組織の再編成を行っている場合)、ヘルスチェックは週に 1 度実行します。

環境の変更に応じて、ヘルスチェックの頻度を調整します。ヘルスチェックの実行頻度に影響する要素を次に示します。

- ◆ パーティションとレプリカの数
- ◆ サーバを保持しているレプリカの安定度
- ◆ eDirectory パーティション内の情報量
- ◆ オブジェクトのサイズと複雑さ
- ◆ 以前の DSRepairs 内のエラー数

ヘルスチェックを実行すると、所有する権利に基づき、iMonitor がすべてのサーバから情報を集めます。なお、ヘルスチェックレポートを実行すると、ネットワークトラフィックが発生し、ディスク容量を消費する場合がありますことに注意してください。

## ヘルスチェックの概要

完全なヘルスチェックでは、次の情報がチェックされます。

- ◆ eDirectory のバージョン  
同一バージョンの NetWare 上で異なるバージョンの NDS や eDirectory を実行していると、同期に問題が発生する場合があります。NDS または eDirectory のバージョンが古い場合は、最新のソフトウェアパッチを [Novell Directory Services Patches and Files \(http://support.novell.com/filefinder/5069/index.html\)](http://support.novell.com/filefinder/5069/index.html) からダウンロードしてください。
- ◆ 時刻の同期  
すべての eDirectory サーバは、正確な時刻を維持する必要があります。タイムスタンプが各オブジェクトおよびプロパティに割り当てられ、これによりオブジェクトおよびプロパティの更新が正しい順序で行われます。eDirectory では、タイムスタンプを使用して、同期が必要なレプリカを判別します。
- ◆ 同期の許容範囲  
インバウンドやアウトバウンドのデータ変更により同期を行ってから経過した期間で、どれだけのデータが未処理となっているかなどをチェックします。
- ◆ バックグラウンド処理  
プロセスはさまざまなタスクを実行しますが、その中には変更の複製およびシステム情報の保守があります。
  - ◆ 外部参照
  - ◆ 破損通知
  - ◆ eDirectory スキーマ

これらのチェックを実行するための詳細な手順については、次のセクション「[iMonitor を使用した eDirectory のヘルスチェック](#)」を参照してください。

## iMonitor を使用した eDirectory のヘルスチェック



環境設定によっては、eDirectory サーバのヘルスチェックを iMonitor の次の 2 つの方法のいずれかを使用して実行できます。

- ◆ [ナビゲータフレームを使用する](#)
- ◆ [アシスタントフレームを使用する](#)

## ナビゲータフレームを使用する

- 1 iMonitor へアクセスする

173 ページの「iMonitor へアクセスする」を参照してください。

- 2 ナビゲータフレームで、[レポート] アイコン  をクリックします。
- 3 アシスタントフレームで、[レポート設定] リンクをクリックします。  
データフレームに、実行可能レポートリストが表示されます。
- 4 [レポートの設定] アイコン  をクリックして、必要なサーバ情報を表示させます。  
データフレームに、サーバ情報レポートが表示されます。このレポートを使用して、レポートに必要なオプションを選択します。
- 5 [ヘルスのサブレポート] チェックボックスをオンにします。
- 6 指定した間隔でレポートを実行するには、データフレームの[レポートのスケジュール] セクションで、必要なオプションを選択します。  
**重要:** スケジュールされたレポートを実行した場合、Public ユーザとして実行されるため、認証されたユーザとして実行するより少ない情報しか得られない場合があります。
- 7 [レポートの実行] をクリックして、レポートを開始します。

## アシスタントフレームを使用する

- 1 iMonitor へアクセスする

173 ページの「iMonitor へアクセスする」を参照してください。

- 2 アシスタントフレームで、[エージェントヘルス] をクリックします。


iMonitor が情報を取得するサーバ(接続先のサーバとは限りません)のヘルスチェック情報が、データフレームに表示されます。

## レポート情報の検討

レポートが生成されたら、データフレームにレポート結果が表示されます。ツリー内に正常に動作していないサーバがある場合、レポートは次の 3 つのカテゴリに分けられます(グループ化では最も正常に動作していないサーバが最初になります)。

- ◆ 警告のあるサーバ
- ◆ 疑わしいサーバ
- ◆ 正常なサーバ

警告のあるサーバや疑わしいサーバがない場合は、これらのカテゴリは表示されません。

正常に動作していないサーバがある場合は、そのサーバの横の [エージェントヘルスのサブレポート] リンク  をクリックします。オンラインのコンテキストヘルプを使用して、問題を解決します。このヘルプは、個々のオプションの意味、それが重要である理由、問題の解決方法、範囲の調整方法、およびヘルスチェックに追加するオプションがあるかどうかを確認するのに役立ちます。

**重要:** 警告のあるサーバがある場合、その問題を解決することを強くお勧めします。疑わしいサーバについても、評価することをお勧めします。

## 詳細情報

eDirectory の正常な動作を維持するために使用するツールおよび技術については、『Novell Certified Directory Engineer Course 991: Advanced eDirectory Tools and Diagnostics』に記載されています。このコースでは、次の方法について学習します。

- ◆ eDirectory ヘルスチェックの実行方法
- ◆ eDirectory の正しい操作方法
- ◆ eDirectory の問題の適切な診断、トラブルシューティング、および解決の方法
- ◆ eDirectory トラブルシューティングのツールおよびユーティリティの使用方法

このコースの詳細については、[Novell Training Services Web サイト \(http://www.novell.com/training/index.html\)](http://www.novell.com/training/index.html) を参照してください。

## 監視のためのリソース

Novell DSTrace ユーティリティは、NetWare、Windows NT、Linux、Solaris、AIX および HP-UX 上で動作します。このツールは、eDirectory の膨大なリソースを監視するのに役立ちます。DSTrace の詳細については、次を参照してください。

- ◆ [184 ページの「トレースを環境設定する」](#)
- ◆ [Looking Into the Directory Services Trace \(DSTrace\) Options \(http://developer.novell.com/research/sections/netmanage/dirprimer/2001/august/spv.htm\)](http://developer.novell.com/research/sections/netmanage/dirprimer/2001/august/spv.htm)
- ◆ [More on Using the DSTrace Command \(http://developer.novell.com/research/sections/netmanage/dirprimer/2001/septembe/p010901.htm\)](http://developer.novell.com/research/sections/netmanage/dirprimer/2001/septembe/p010901.htm)

また、eDirectory 環境用の他の管理ソリューションを提供するサードパーティの製品も使用できます。詳細については、次の Web サイトを参照してください。

- ◆ [BindView \(http://www.bindview.com\)](http://www.bindview.com)
- ◆ [Blue Lance \(http://www.bluelance.com\)](http://www.bluelance.com)
- ◆ [NetPro\\* \(http://www.netpro.com\)](http://www.netpro.com)

弊社のパートナーが提供していない eDirectory の特性をモニタしたり、監視する必要がある場合、Novell Consulting Services が Novell Event System を使用した評価と監視のカスタマイズをお手伝いいたします。

## ハードウェアのアップグレードやサーバの交換

このセクションでは、ハードウェアをアップグレードまたは交換する際に、特定のサーバ上の eDirectory を移す、または保護するための情報について説明します。ここでの説明は、[381 ページの「Novell eDirectory のバックアップと復元」](#)の情報に基づいています。

Backup eDirectory Management Tool を使用すれば、次の操作を行うための eDirectory 情報を作成できます。

- ◆ [461 ページの「サーバを交換しないでハードウェアまたは記憶デバイスを計画的にアップグレードする」](#)
- ◆ [464 ページの「サーバの計画的な交換」](#)



## サーバを交換しないでハードウェアまたは記憶デバイスを計画的にアップグレードする

記憶デバイスやRAMなどのハードウェアのアップグレードを計画している場合、Backup eMtool を使用して eDirectory およびファイルシステムのコールドバックアップを行います。これにより、サーバの eDirectory 識別情報とファイルシステムデータが保護されます。このバックアップには、次の利点があります。

- ◆ 記憶デバイスを交換する場合、バックアップによって古い記憶デバイスから新しい記憶デバイスに情報を移すことができます。
- ◆ eDirectory が格納されたディスクパーティションまたはディスクボリュームを含む記憶デバイスを交換する場合、このバックアップ情報を用いれば、復元プロセスを使用して eDirectory データベースを新しい記憶デバイス上に再構築できます。
- ◆ eDirectory のコールドバックアップを実行し、その後でデータベースをクローズしておけば、バックアップ後のデータベースの変更を心配することなくハードウェアをアップグレードし、データベースを移すことができます。
- ◆ 何か問題が発生した場合、バックアップを使用して復元できます。

eDirectory のコールドバックアップを実行する場合、オプションを使用してサーバ上の eDirectory をロックし、使用不可にする必要があります。これにより、バックアップ後のデータ変更を防げます。このサーバと通信している他のサーバからは、このサーバは停止しているように見えます。通常サーバに送信される eDirectory 情報は、そのサーバと再び通信できるようになるまで、ツリー内の別のサーバに保存されます。保存された情報は、サーバがオンライン状態に戻ったときに、サーバを同期するために使用されます。

**注：**eDirectory ツリー内の他のサーバは、このサーバがすぐにオンライン状態に戻ると期待しているため、アップグレードをすばやく完了させ、できるだけ早くサーバ上の eDirectory データベースをオープンする必要があります。

ハードウェアのアップグレードを計画的に実行するには、次の手順に従います。

- 1 アップグレードによりサーバに問題が発生するかもしれないと心配なら、必要に応じて、使用する別のコンピュータを準備するとよいでしょう。

464 ページの「1. サーバ交換の準備」を参照してください。

- 2 eDirectory データベースのコールドバックアップを実行し、その後でデータベースをクローズし、ロックしたままにしておくには、eMBox クライアントコマンドを次のように使用します。NICI を使用する場合は、`-e` オプションを使用してセキュリティファイルもバックアップします。

```
backup -f backup_filename_and_path  
-l log_filename_and_path -e -t -c -o -d
```

NICI を使用する場合、`-e` スイッチを使用して NICI ファイルをバックアップします。(eMBox クライアントとスイッチの使用についての詳細は、412 ページの「eMBox クライアントによる手動バックアップ」および 423 ページの「バックアップ / 復元のコマンドラインオプション」を参照してください)。

これで、eDirectory データベースはロックされました。手順を完了するまでは、サーバ上で新たなデータ変更が実行されないように、データベースをロックしたままにしておく必要があります。

サーバが使用できない時間を最小限に抑えるために、以降の手順をすばやく完了させます。

- 3 お好みのバックアップツールを使用して、ファイルシステムをバックアップします。(Netware の場合、SMS™ が使用できます)。

データベースをバックアップした後で、ファイルシステムのバックアップを行うのは重要です。これにより、eDirectory バックアップファイルが、他のファイルシステムと一緒にテープに保存されます。

- 4 サーバを停止させ、ハードウェアを交換します。
- 5 ハードウェアを交換した後で、ハードウェア変更の種類に応じた以下の手順を実行します。

ハードウェア変更の種類/	実行する手順
記憶デバイスに変更がない場合	サーバを起動し、データベースのロック解除を行います。
記憶デバイスの交換を行ったが、eDirectory が格納されたディスクパーティション/ボリュームに変更はない場合	<ol style="list-style-type: none"> <li>1. サーバと eDirectory を起動します。</li> <li>2. 交換した記憶デバイス上にあったディスクパーティション/ボリュームのファイルシステムだけを復元します。</li> <li>3. eDirectory データベースのロックを解除します。</li> </ol>
NetWare 以外のオペレーティングシステム環境の eDirectory が格納された記憶デバイスを交換した	<ol style="list-style-type: none"> <li>1. 必要に応じてオペレーティングシステムをインストールします。</li> <li>2. 記憶デバイスの変更により影響を受けたディスクパーティション上に、ファイルシステムを復元します。</li> <li>3. 新しい記憶デバイス上の、新しい一時的なツリー内に、eDirectory をインストールします。</li> <li>4. バックアップから eDirectory を復元します (元のツリー内に配置されます)。このとき、復元した後で eDirectory がクローズされ、ロックされたままになるようにオプションを指定します。次のようなコマンドを使用します。  <pre>restore -r -f backup_filename_and_path -l log_filename_and_path -e</pre> <p>NICI ファイルをバックアップしてあった場合は、-e オプションを使用します。インクルードファイルに列挙されたファイルをバックアップしてあった場合は、-u オプションを追加します。</p> </li> <li>5. eDirectory データベースのロックを解除します。</li> <li>6. NICI セキュリティファイルを復元した場合は、復元を完了した後で、サーバを再起動してセキュリティシステムを再初期化します。</li> <li>7. サーバが通常どおりに応答するか、チェックします。</li> </ol> <p>ConsoleOne<sup>®</sup> を使用して、サーバとその同期をチェックします。ログインスクリプトと印刷が正しく動作することを確認します。</p> <ol style="list-style-type: none"> <li>8. このサーバでロールフォワードログを使用していた場合、復元を完了した後で、ロールフォワードログ設定を再作成します。ロールフォワードログを有効にしているから、改めてフルバックアップも取る必要があります。</li> </ol> <p>復元した後は、設定がデフォルトの状態にリセットされます。つまり、ロールフォワードログがオフになっています。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。</p>

ハードウェア変更の種類/	実行する手順
Netware 上の eDirectory と SYS: ポリリュームが含まれた記憶デバイスを交換した	<p data-bbox="806 163 1459 368">NetWare 上のファイルシステムデータを復元する場合、ファイルシステム権の保持に関わる問題があることを認識しておく必要があります。eDirectory を復元した後で、ファイルシステムを復元する必要があります。また、<a href="#">396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」</a>で説明している追加の手順を実行すると良いでしょう。</p> <ol data-bbox="806 389 1459 741" style="list-style-type: none"> <li>1. 新しい記憶デバイスに NetWare と eDirectory をインストールし、一時的な新しいツリー内に新しい SYS: ポリリュームを作成します。</li> <li>2. 新しい SYS: ポリリューム上に、バックアップテープからコピーした eDirectory バックアップファイルを配置します。</li> <li>3. バックアップから eDirectory を復元します (元のツリー内に配置されます)。このとき、復元した後で eDirectory がクローズされ、ロックされたままになるようにオプションを指定します。次のようなコマンドを使用します。  <pre data-bbox="837 687 1264 741">restore -r -f backup_filename_and_path -l log_filename_and_path -e</pre> </li> </ol> <p data-bbox="837 774 1459 889">NICI ファイルをバックアップしてあった場合は、-e オプションを使用します。インクルードファイルに列挙されたファイルをバックアップしてあった場合は、-u オプションを追加します。</p> <ol data-bbox="806 909 1459 1155" style="list-style-type: none"> <li>4. 記憶デバイスの交換によって影響を受けたすべてのポリリュームのファイルシステムを復元します。</li> <li>5. eDirectory データベースのロックを解除します。</li> <li>6. NICI セキュリティファイルを復元した場合は、復元を完了した後で、サーバを再起動してセキュリティシステムを再初期化します。</li> <li>7. サーバが通常どおりに応答するか、チェックします。</li> </ol> <p data-bbox="837 1187 1459 1272">iMonitor を使用して、サーバとその同期をチェックします。ログインスクリプトと印刷が正しく動作することを確認します。</p> <ol data-bbox="806 1292 1459 1397" style="list-style-type: none"> <li>8. このサーバでロールフォワードログを使用していた場合、復元を完了した後で、ロールフォワードログ設定を再作成します。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。</li> </ol> <p data-bbox="837 1429 1459 1568">復元した後は、設定がデフォルトの状態にリセットされません。つまり、ロールフォワードログがオフになっています。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起こる可能性があるためです。</p>

サーバが通常どおりに応答しない場合、次にいずれかの方法によって回復する必要があります。

- ◆ 変更前のハードウェア環境設定が機能していたことから、これを再作成します。
- ◆ 作成した eDirectory バックアップとファイルシステムを使用して、このサーバの識別情報を別のコンピュータに移します。[464 ページの「サーバの計画的な交換」](#)を参照してください。

## サーバの計画的な交換

次に示す手順で、サーバの eDirectory アイデンティティとファイルシステムデータを別のコンピュータ上に移すことで、実際にサーバを置き換えます。ここでは、古いサーバをサーバ A とし、それに置き換わるサーバをサーバ B としています。

Backup eMTool を使用して eDirectory のコールドバックアップ (データベースをクローズした状態でのバックアップ) を実行し、さらにお好みのツールを使用してファイルシステムのバックアップをして、サーバの交換に備えます。このバックアップ情報を用いれば、復元プロセスを使用して、新しいコンピュータ上にサーバを再構築できます。

eDirectory のコールドバックアップを実行する場合、オプションを使用してサーバ A 上の eDirectory をロックし、使用不可にする必要があります。これにより、バックアップ後のデータ変更を防げます。このサーバと通信している他のサーバからは、このサーバは停止しているように見えます。通常サーバに送信される eDirectory 情報は、そのサーバと再び通信できるようになるまで、ツリー内の別のサーバに保存されます。保存された情報は、新しいコンピュータであるサーバ B 上で、サーバがオンライン状態に戻ったときに同期するために使用されます。

**注：**eDirectory ツリー内の他のサーバは、このサーバがすぐにオンライン状態に戻ることを期待しているため、できるだけ早く交換してサーバに eDirectory 情報を復元する必要があります。

サーバを置き換えるための手順の概要を次に示します。

1. 交換する際のサーバ A の停止時間を短くするには、[464 ページの「1. サーバ交換の準備」](#)で説明しているようにサーバ B にオペレーティングシステムをインストールするなどして、交換前にできるだけサーバ B の準備を整えておきます。
2. [465 ページの「2. eDirectory のバックアップを作成する」](#)の説明に従って、サーバ A の eDirectory とシステムファイルをバックアップします。
3. [466 ページの「3. サーバ交換における eDirectory 情報の復元」](#)の説明に従って、サーバ B に情報を移します。

### 1. サーバ交換の準備

次に示すサーバ A とサーバ B のチェックリストを使用して、サーバ A を交換する準備ができているか確認します。開始する前にサーバ B の準備をしておけば、あるコンピュータから別のコンピュータへ転送する間のサーバの停止時間を減らせます。

#### サーバ A の準備

- サーバ A に最新のバージョンのオペレーティングシステムがインストールされていることを確認します。
- Tree パーティションのマスタを保持しているサーバで DSRepair を実行し、さらに時刻同期を実行して、サーバ A のツリーが正常に機能していることを確認します。
- サーバ A のデータベースで DSRepair を実行します。サーバ A が完全に同期されていることを確認します。

#### サーバ B の準備

- 最新バージョンのオペレーティングシステムをインストールします。このオペレーティングシステムは、サーバ A のものと同じである必要があります。
- サーバ B を新しい一時的なツリーに配置し、eDirectory をインストールします。

([466 ページの「3. サーバ交換における eDirectory 情報の復元」](#)の過程で eDirectory を復元するには、サーバ B をサーバ A が配置されていた元のツリー内に配置します)。

- (NetWare の場合のみ) サーバの交換でファイルシステムデータを復元する場合、ファイルシステム権の保持に関わる問題があることを認識しておく必要があります。ファイルシステムの復元の前に、eDirectory の復元を計画する必要があります。また、396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」で説明している追加の手順を実行すると良いでしょう。

次のセクション「2. eDirectory のバックアップを作成する」に示す手順に続きます。

## 2. eDirectory のバックアップを作成する

サーバ交換の前に、eDirectory のバックアップを作成する必要があります。464 ページの「1. サーバ交換の準備」が完了した後は eMBox Client を使用し、バックアップの後でデータベースを使用不可にしてロックする詳細オプションを設定して、サーバ A 上の eDirectory データベースのコールドバックアップを実行します。

eDirectory のコールドバックアップ(データベースがクローズ中のバックアップ)を作成し、その後でデータベースをクローズのままにしておくには、次の手順に従います。

- 1 464 ページの「1. サーバ交換の準備」が完了していることを確認します。
- 2 次に示すような eMBox クライアントのコマンドで、-c、-o、および -d スイッチを使用して、サーバ A 上の eDirectory データベースのコールドバックアップを実行し、完了した後はデータベースをクローズしてロックしたままにします。

```
backup -f backup_filename_and_path  
-l log_filename_and_path -e -t -c -o -d
```

NICI を使用する場合、-e スイッチを使用して NICI ファイルをバックアップします。(eMBox クライアントとスイッチの使用についての詳細は、412 ページの「eMBox クライアントによる手動バックアップ」および 423 ページの「バックアップ / 復元のコマンドラインオプション」を参照してください)。

サーバ A の eDirectory データベースがロックされます。データベースをサーバ B 上に復元しツリー内に戻すまでは、サーバ上で新たなデータ変更が実行されないように、データベースをロックしたままにしておく必要があります。

サーバアップグレードまたはサーバ交換の残りの手順を迅速に完了させ、サーバが使用できない時間を最小限に抑えます。

- 3 サーバ A のファイルシステムのフルバックアップを作成します。(NetWare では、SMS が使用できます)。

データベースをバックアップした後で、ファイルシステムのバックアップを行うのは重要です。これにより、eDirectory バックアップファイルが、残りのファイルシステムと一緒にテープに保存されます。

SMS の使用方法の詳細については、『*Storage Management Services Administration Guide(Storage Management Services 管理ガイド)* (<http://www.novell.com/documentation/japanese/nw65/smsadmin/data/hjc2z4tu.html>)』を参照してください。

- 4 サーバ A 上の eDirectory データベースをロックし、サーバ A をネットワークから外します。

続いて 466 ページの「3. サーバ交換における eDirectory 情報の復元」の手順を実行します。

### 3. サーバ交換における eDirectory 情報の復元

サーバ A の eDirectory 識別情報およびファイルシステムをサーバ B に移すには、次の手順に従います。

- 1 464 ページの「1. サーバ交換の準備」および 465 ページの「2. eDirectory のバックアップを作成する」が完了していることを確認します。
- 2 サーバ B が起動し、eDirectory が実行されていることを確認します。
- 3 復元により、サーバ A の eDirectory 識別情報およびファイルシステムを次の手順でサーバ B に移します。

**3a** サーバ A の eDirectory コールドバックアップファイルをサーバ B にコピーします。

サードパーティのファイル圧縮ツールは圧縮性能が良いので、そのようなツールを使用した場合、バックアップファイルはとても小さくなる場合があります。これにより、ファイルのコピーを早くできる場合があります。

**3b** 複製した eDirectory のバックアップファイルを使用して、サーバ A の eDirectory データベースをサーバ B 上に復元します。それには、eMBox コマンドラインクライアントで、次のようなコマンドを使用します。

```
restore -r -f backup_filename_and_path  
-l log_filename_and_path -e
```

NICI を使用する場合、`-e` スイッチを使用して NICI ファイルを復元します。インクルードファイルに列挙されたファイルをバックアップしてあった場合は、`-u` オプションを追加します。(eMBox クライアントとスイッチの使用についての詳細は、420 ページの「eMBox クライアントによるバックアップファイルの復元作業」および 423 ページの「バックアップ / 復元のコマンドラインオプション」を参照してください)。

復元にはロールフォワードログを含める必要はありません。なぜなら、コールドバックアップを実行し、その後でデータベースをクローズしてあるからです。データベースではどのようなトランザクションも実行されていません。データベースはクローズされ、バックアップ以降にはロールフォワードログは作成されていません。

**重要 :** NetWare では、ファイルシステムを復元する前に eDirectory を復元することが特に重要です。これにより、トラスティ割り当ておよび権利が、ファイルシステムデータの復元の後で保持されます。詳細については、396 ページの「NetWare のファイルシステムデータを復元する際のアクセス権の保存」を参照してください。

**3c** バックアップされたサーバ A のファイルシステムデータをサーバ B に移します。

- 4 (NetWare の場合のみ) `autoexec.ncf` で、サーバ B の IP アドレスとサーバ名を、サーバ A のものにリネームします。
- 5 NICI を使用している場合は、サーバを再起動して NICI を再初期化し、復元された NICI セキュリティファイルが使用されるようにします。
- 6 eDirectory データベースのロックを解除します。
- 7 復元が完了した後は、サーバ B がサーバ A の識別情報を正しく引き継ぎ、通常どおりに応答しているかチェックします。ConsoleOne を使用してサーバとその同期をチェックします。ログインスクリプト、印刷、および NICI セキュリティが正常に機能することを確認します。

サーバの応答が通常どおりなら、サーバの交換は完了です。これで、サーバ A から eDirectory をアンインストールして eDirectory 識別情報を削除し、このコンピュータを別の目的に使用できます。サーバ A をネットワークに戻すのは、eDirectory を削除した後に行ってください。そうしないと、eDirectory の同期でネットワークが混乱してしまいます。なぜなら、サーバ A とサーバ B の同じ識別情報により、競合が発生するためです。

- 8 (状況によって実行)** このサーバでロールフォワードログを使用していた場合、復元を完了した後で、ロールフォワードログ設定を作成し直します。ロールフォワードログを有効にしてから、改めてフルバックアップも取る必要があります。

復元した後は、設定がデフォルトの状態にリセットされます。つまり、ロールフォワードログがオフになっています。フルバックアップが改めて必要となるのは、スケジュールに従って次に無人でのフルバックアップが取られるまでに、再び障害が起る可能性があるためです。

サーバ B が正常に動作せず、サーバ A の識別情報およびファイルシステムを直ちに使用できるようにする必要がある場合は、次を実行します。

- 1 サーバ B のネットワークケーブルを抜くか、またはサーバを停止します。
- 2 サーバ A をネットワークに再接続し、起動してから、eDirectory データベースをオープンします。  
DSRepair の実行を要求するシステムメッセージを無視します。
- 3 サーバ B から eDirectory を削除し、再度アップグレードを試みます。

## ハードウェア障害後の eDirectory の復元

eDirectory が配置されたディスクパーティションまたはディスクボリュームを含むハードディスクの障害は、サーバから eDirectory が削除された状態と同じです (幸いにも複数サーバ環境では、1 つのサーバがダウンしても、そのレプリカリング内の残りのサーバが正常に稼働していれば問題はありません)。

eDirectory が格納されたディスクパーティションまたはディスクボリュームの障害の後で eDirectory を復元するには、[401 ページの「復元処理の準備」](#) および [409 ページの「iManager によるバックアップファイルの復元作業」](#) (または、[420 ページの「eMBox クライアントによるバックアップファイルの復元作業」](#)) で説明している手順に従って、バックアップファイルから復元します。

ハードディスクの新規インストールでは、製造元から提供されている指示に従って、サーバのハードディスクが動作することを検証します。新しいハードディスクには、少なくとも置き換えられる元のドライブと同じ記憶容量が必要です。ローカルサーバ情報のファイルを使用して、環境設定情報を確認します。

**注:** サーバのバックアップファイルがない場合は、Xbrowse ツールを使用して eDirectory に照会し、サーバ情報を回復します。この作業は、サーバオブジェクトやその関連オブジェクトをツリーから削除する前に実行する必要があります。XBrowse およびその他の情報は『[Technical Information Document #2960653 \(http://support.novell.com/servlet/tidfinder/2960653\)](http://support.novell.com/servlet/tidfinder/2960653)』にあり、Novell サポートから入手できます。



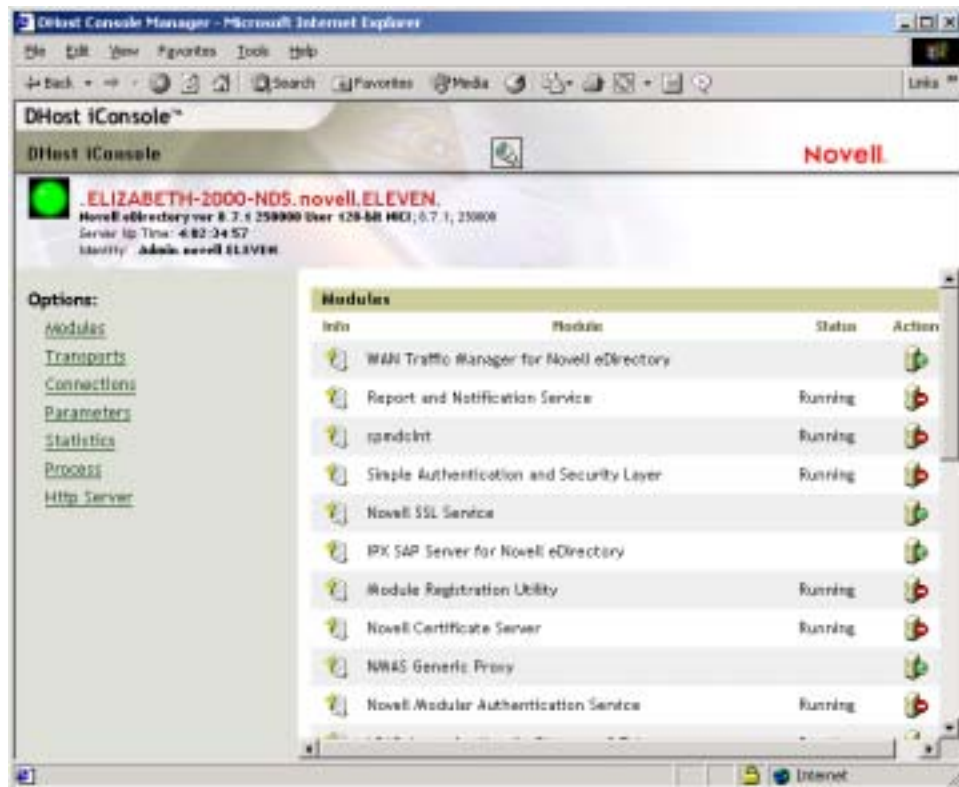


# 16 DHost iConsole Manager

DHost iConsole Manager は Web ベースのブラウザを使用した管理ツールで、次のことが行えます。

- ◆ DHost モジュールの管理
- ◆ DHost 環境設定パラメータの照会
- ◆ DHost 接続情報の表示
- ◆ スレッドプール統計情報の表示
- ◆ DHost プロトコルスタックマネージャにより登録されたプロトコルに関する詳細情報の表示

図 35 DHost iConsole Manager



DHost iConsole Manager は診断およびデバッグツールとしても使用でき、eDirectory サーバが正常に動作していない場合に HTTP サーバにアクセスできます ( 詳細については、[476 ページの「SAdmin パスワードを設定する」](#)を参照してください)。

この章では、次の情報について説明します。

- ◆ 470 ページの「DHost について」
- ◆ 471 ページの「DHost iConsole の実行」
- ◆ 472 ページの「eDirectory モジュールの管理」
- ◆ 474 ページの「DHost 情報の照会」
- ◆ 476 ページの「プロセススタック」
- ◆ 476 ページの「SAdmin パスワードを設定する」

## DHost について

Windows、Solaris、Linux、AIX、および HP-UX 用の Novell® eDirectory ソフトウェアは、eDirectory for NetWare® と同じコアコードに基づいて作成されています。Windows および UNIX 版の eDirectory が別のバージョンの eDirectory と適切にやり取りを行うには、NCP™(NetWare Core Protocol™) サービスのサブセットをサポートする必要があります。このサブセットを操作するプログラムは DHost と呼ばれます。DHost は eDirectory の下で動作し、NetWare オペレーティングシステムがネイティブで提供する機能を NetWare 以外のプラットフォームで提供します。

DHost は、次の NetWare 指向のサービスを提供します。

サービス	説明
NCP エンジン	<p>パケットベースのプロトコルで、クライアントが NetWare サーバとの間でリクエストの送信や応答の受信をできるようにします。</p> <p>詳細については、『<a href="http://developer.novell.com/ndk/doc/ncp/ncp__enu/data/hc4lztgy.html">NetWare Core Protocols (http://developer.novell.com/ndk/doc/ncp/ncp__enu/data/hc4lztgy.html)</a>』を参照してください。</p>
ウォッチドッグ	<p>ワークステーションが NetWare サーバに接続された状態であることを確認するために使用するパケットです。</p> <p>詳細については、『<a href="http://www.novell.com/documentation/japanese/nw65/ipx_enu/data/h0cufuir.html">Watchdog Packet Spoofing (http://www.novell.com/documentation/japanese/nw65/ipx_enu/data/h0cufuir.html)</a>』参照してください。</p>
接続テーブル	<p>NetWare サーバに付随するあらゆるプロセス、プリントサーバ、アプリケーション、ワークステーション、またはその他のエンティティに割り当てられる固有の番号です。この番号は、接続が行われるごとに異なる可能性があります。接続番号は、ネットワークセキュリティの実装やネットワークアカウントに使用されます。この番号は、ファイルサーバ接続テーブル内でのオブジェクトの場所を反映しています。さらにこの番号を使用すると、ネットワークにログインしたオブジェクトに関する情報の識別や取得が容易になります。</p>
イベントシステム	<p>個々のサーバのアクティビティを監視する手段をアプリケーションに提供します。</p>
スレッドプール	<p>独立したエンティティとして実行され、システムソフトウェアによってスケジューラされる一連の命令です。</p>
NCP 拡張	<p>サーバアプリケーション開発者が、NetWare OS に NCP として実装する NLM™ ソフトウェアを作成できるようにします。</p> <p>詳細については、『<a href="http://developer.novell.com/ndk/doc/ncp/index.html?page=ndk/doc/ncp/ncp__enu/data/a1wftl8.html">NCP Extension Concepts (http://developer.novell.com/ndk/doc/ncp/index.html?page=ndk/doc/ncp/ncp__enu/data/a1wftl8.html)</a>』を参照してください。</p>

サービス	説明
メッセージ処理	ドキュメントを圧縮または凝縮した形式、またはドキュメントの要約で、大きなドキュメントの電子指紋として機能します。メッセージ処理は、個々のドキュメントに固有のデジタル署名を作成するために使用されます。

## DHost iConsole の実行

- ◆ 471 ページの「[NetWare で DHost iConsole を実行する](#)」
- ◆ 471 ページの「[Windows で DHost iConsole を実行する](#)」
- ◆ 472 ページの「[Linux、Solaris、AIX、および HP-UX で DHost iConsole を実行する](#)」

### NetWare で DHost iConsole を実行する

NetWare では、NetWare リモートマネージャを使用して DHost iConsole にアクセスできます。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで `httpstk.nlm` を実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**`http://server's TCP/IP address:port`**

次に例を示します。

`http://137.65.123.11:8008`

**注：**デフォルトの代替ポート番号は 8008 です。NetWare の [リモートマネージャの環境設定] ページでこの値を変更した場合は、その新しいポート番号を入力します。

サーバ名と IP アドレスの解決のために DNS(Domain Name Service) をネットワークにインストールしてある場合は、IP アドレスの代わりにサーバの DNS 名を入力することもできます。

- 3 ユーザ名、コンテキスト、パスワードを指定します。

### Windows で DHost iConsole を実行する

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**`http://server.name:port/dhost`**

たとえば、次のように入力します。

`http://MyServer:80/dhost`

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

`http://137.65.135.150:80/dhost`

- 3 ユーザ名、コンテキスト、パスワードを指定します。

## Linux、Solaris、AIX、および HP-UX で DHost iConsole を実行する

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost





- 3 ユーザ名、コンテキスト、パスワードを指定します。

## eDirectory モジュールの管理

DHost iConsole の [モジュール] ページでは、使用可能な eDirectory サービスとその状態についての情報が表示されます。また、この [モジュール] ページからこれらのサービスを開始または停止 (ロードまたはアンロード) できます。

ロードやアンロードができるのは、LDAP、SNMP、HTTPSTK などの非対話型モジュールだけです。

[モジュール] ページには、次に示す属性があります。

属性	説明
情報	 をクリックして、モジュールの説明、ファイル名、モジュールハンドル、属性、および選択したモジュールの共有オブジェクト名を表示させます。
モジュール	モジュール名を表示します。
ステータス	モジュールが実行されているかどうかを表示します。
アクション	モジュールが実行可能かどうかを示します。モジュールのステータスは、次の 3 つのうちのどれかになります。  は、モジュールがシステムモジュールで、アンロードできないことを示します。  は、モジュールがロード可能で、ロードの準備ができていることを示します。  は、モジュールが実行中であることを示します。

- ◆ [473 ページの「NetWare でモジュールをロードまたはアンロードする」](#)
- ◆ [473 ページの「Windows でモジュールをロードまたはアンロードする」](#)
- ◆ [473 ページの「Linux、Solaris、AIX、および HP-UX でモジュールをロードまたはアンロードする」](#)

Novell iManage を使用した eDirectory サービスのロードやアンロードについての詳細は、[168 ページの「eDirectory Service Manager」](#) を参照してください。

## NetWare でモジュールをロードまたはアンロードする

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server's TCP/IP address:port**

次に例を示します。

http://137.65.123.11:8008

注：デフォルトの代替ポート番号は 8008 です。NetWare の [リモートマネージャの環境設定] ページでこの値を変更した場合は、その新しいポート番号を入力します。

サーバ名と IP アドレスの解決のために DNS(Domain Name Service) をネットワークにインストールしてある場合は、IP アドレスの代わりにサーバの DNS 名を入力することもできます。

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 アプリケーションの管理リストで [List Modules(モジュールを一覧)] をクリックします。
- 5 モジュールをロードするには、名前を入力して [Load Module(モジュールをロード)] をクリックします。

モジュールが実際にロードされたかどうかの確認が必要な場合は、[モジュールロード用のシステムコンソールの表示] チェックボックスをオンにします。

## Windows でモジュールをロードまたはアンロードする

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。



**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 [モジュール] をクリックします。
- 5 モジュールをロードするには  をクリックし、アンロードするには  をクリックします。

## Linux、Solaris、AIX、および HP-UX でモジュールをロードまたはアンロードする

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。



**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

```
http://137.65.135.150:80/dhost
```

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 [モジュール] をクリックします。
- 5 モジュールをロードするには  をクリックし、アンロードするには  をクリックします。

## DHost 情報の照会

DHost iConsole マネージャを使用すれば、次に示す情報を照会できます。

- ◆ 環境設定パラメータ
- ◆ PSTACK マネージャにより登録されたプロトコル
- ◆ 接続プロパティ
- ◆ スレッドプールの概要

## 環境設定パラメータを表示する

環境設定パラメータは、UNIX プラットフォームに特有のものです。

DHost iConsole マネージャで、[パラメータ] をクリックします。詳細については、[472 ページの「Linux、Solaris、AIX、および HP-UX で DHost iConsole を実行する」](#)を参照してください。

環境設定パラメータには、次の情報が表示されます。

オプション	説明
パラメータ名	環境設定パラメータの名前を表示します。
デフォルト値	環境設定パラメータのデフォルト値を表示します。
設定値	現在の設定値を表示します。
最小値	パラメータに設定できる最小値を表示します。
最大値	パラメータに設定できる最大値を表示します。
タイプ	パラメータに設定できる値のタイプを表示します。

詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「[環境設定パラメータ](#)」を参照してください。

## プロトコル情報を表示する

DHost iConsole マネージャで、[トランスポート] をクリックします。

次のプロトコル情報が表示されます。

- ◆ ID
- ◆ プロトコル
- ◆ トランスポート

## 接続プロパティを表示する

DHost iConsole マネージャで、[接続] をクリックします。

次の接続プロパティが表示されます。

- ◆ 接続
- ◆ フラグ
- ◆ 識別子
- ◆ 表示名
- ◆ トランスポート
- ◆ 認証名
- ◆ SEV 回数
- ◆ 最終アクセス
- ◆ ロック状態

## スレッドプールの統計情報を表示する

DHost iConsole マネージャで、[統計情報] をクリックします。

次のスレッドプール統計情報が表示されます。

- ◆ Spawned Threads( 生成スレッド )
- ◆ Dead Threads( 停止スレッド )
- ◆ Idle Threads( アイドルスレッド )
- ◆ Worker Thread( 動作スレッド )
- ◆ Peak Worker Thread( ピーク動作スレッド )
- ◆ Ready for Work Thread( 動作待機スレッド )
- ◆ Ready Queue Peak Worker Threads( 待機キューピーク動作スレッド )
- ◆ Ready Queue Max Wait Time( 待機キュー最大待ち時間 )
- ◆ Schedule Delay Minimum Time( 最小スケジュール遅延時間 )
- ◆ Schedule Delay Maximum Time( 最大スケジュール遅延時間 )
- ◆ Schedule Delay Average Time( 平均スケジュール遅延時間 )
- ◆ Waiting For Work( 動作待ち )
- ◆ Peaking Waiting For Work( ピーク動作待ち )

## プロセススタック

プロセススタックには、DHost のプロセス空間で現在実行されているすべてのスレッドのリストが含まれます。スレッドについての詳細な情報を表示するには、スレッド ID をクリックします。この機能は主として Novell の技術者およびサポート担当者がローレベルのデバッグツールとして使用するためのものです。

このオプションは、Windows でのみ使用できます。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 [プロセス] をクリックします。
- 5 スレッドのコールスタックを表示するには、スレッド ID をクリックします。

## SAdmin パスワードを設定する

あらかじめ設定された admin ユーザをセットアップできます。これにより、eDirectory がロードされていない場合に、HTTPSTK(HTTP Protocol Stack) にアクセスできるようになります。事前に設定された管理者ユーザ (SAdmin) には、eDirectory 管理者ユーザオブジェクトと同等の権利があります。サーバが、eDirectory が適切に機能していない状態の場合、このユーザとしてサーバにログインし、eDirectory を使用せずに実行できる必要なすべての診断およびデバッグ作業を実行します。

- ◆ [476 ページの「NetWare で SAdmin パスワードを設定する」](#)
- ◆ [477 ページの「Windows で SAdmin パスワードを設定する」](#)
- ◆ [477 ページの「Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する」](#)

## NetWare で SAdmin パスワードを設定する

NetWare リモートマネージャを使用して SAdmin ユーザオブジェクトを有効にし、このオブジェクトのパスワードを設定または変更します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで httpstk.nlm を実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server's TCP/IP address:port**


次に例を示します。

http://137.65.123.11:8008

注：デフォルトの代替ポート番号は 8008 です。NetWare の [リモートマネージャの環境設定] ページでこの値を変更した場合は、その新しいポート番号を入力します。



サーバ名と IP アドレスの解決のために DNS(Domain Name Service) をネットワークにインストールしてある場合は、IP アドレスの代わりにサーバの DNS 名を入力することもできます。

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 [環境設定] ボタン  [Enable Emergency Account(SADMIN User)] の順にクリックし、[パスワードの設定] をクリックします。
- 5 SAdmin パスワードを指定し、次に指定したパスワードを確認入力します。
- 6 [設定] をクリックします。

## Windows で SAdmin パスワードを設定する

DHOST の [リモートマネージャ] ページ (/dhost URL またはルートページからアクセス可能) を使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで dhost.exe を実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 HTTP サーバをクリックしてから、SAdmin パスワードを指定します。
- 5 指定したパスワードを確認入力して、[送信] をクリックします。

## Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する

Solaris、Linux、AIX、または HP-UX で SAdmin パスワードを設定するには、次のいずれかの方法を使用できます。

- ◆ [477 ページの「DHOST リモート管理ページ」](#)
- ◆ [478 ページの「ndsconfig」](#)

### DHOST リモート管理ページ

DHOST の [リモートマネージャ] ページ (/dhost URL またはルートページからアクセス可能) を使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで Novell eDirectory サーバを実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

```
http://137.65.135.150:80/dhost
```

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 HTTP サーバをクリックしてから、SAdmin パスワードを指定します。
- 5 指定したパスワードを確認入力して、[送信] をクリックします。

## ndsconfig

ndsconfig ユーティリティを使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで ndsd を実行している必要があります。

サーバコンソールから、次のように入力します。

```
ndsconfig set http.server.sadmin-pwd=password
```

ここで *password* は、新しい SAdmin パスワードです。

ndsconfig ユーティリティの使用の詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「**ndsconfig ユーティリティパラメータ**」を参照してください。

# 17

## eDirectory Management Toolbox

Novell® eDirectory™ Management Toolbox (eMBox) を使用すれば、すべての eDirectory バックエンドユーティリティに、サーバ上だけでなくリモートからもアクセスできます。

eMBox を Novell iManager とあわせて使用すると、DSRepair、DSMerge、バックアップと復元、サービスマネージャなどの eDirectory ユーティリティに Web ベースでアクセスできます。

**重要** : eMBox タスクを実行するには、iManager を使用して、管理するツリーに役割ベースサービスを設定する必要があります。

すべての機能は、ローカルサーバまたはリモートのいずれからでもコマンドラインクライアントを通じて使用できます。eMBox クライアントを使用して、1 つのサーバまたはワークステーションから複数のサーバに対するタスクを実行できます。

バックアップ、DSRepair、DSMerge、スキーマの操作、および eDirectory Service Manger などのすべての eMTool(eDirectory Management Tool) を実行するためには、eDirectory サーバに eMBox がロードされ、実行されている必要があります。

このセクションでは、次のトピックについて説明します。

- ◆ [479 ページの「eMBox コマンドラインクライアントの使用」](#)
- ◆ [489 ページの「eMBox ログの記録の使用」](#)

### eMBox コマンドラインクライアントの使用

eMBox にアクセスする方法の 1 つは、eMBox の Java コマンドラインクライアントを使用することです。このコマンドラインクライアントには、対話式モードとバッチモードの 2 つのモードがあります。対話式モードでは、eMBox コマンドを一度に 1 つずつ実行します。バッチモードでは、コマンドのグループを自動で実行できます。コマンドラインクライアントにはログサービスがあり、いずれのモードでも使用できます。

コマンドラインクライアントは、Java アプリケーションです。これを実行するには、Java Runtime Environment, Sun JVM 1.3.1 が使用できる必要があります。この Java 実行環境は、eDirectory と同時にインストールされます。また、ファイアウォール越しに管理対象のサーバにアクセスできる必要があります。1 つのサーバまたはワークステーションから、複数のサーバに対してタスクを実行できます。

このセクションでは、次のトピックについて説明します。

- ◆ [480 ページの「コマンドラインヘルプを表示する」](#)
- ◆ [480 ページの「eMBox コマンドラインクライアントを対話式モードで実行する」](#)
- ◆ [484 ページの「eMBox コマンドラインクライアントをバッチモードで実行する」](#)
- ◆ [486 ページの「eMBox コマンドラインクライアントのオプション」](#)
- ◆ [487 ページの「eMBox クライアントを使用してセキュア接続を確立する」](#)
- ◆ [488 ページの「eDirectory ポート番号を確認する」](#)

## コマンドラインヘルプを表示する

eMBox クライアントを実行する前に、eMBox の一般的なコマンドラインヘルプを表示するには、次の操作を実行します。

- ◆ NetWare<sup>®</sup>およびUNIXの場合は、コマンドラインから「edirutil -?」と入力します。
- ◆ Windows の場合は、次を実行します。  
`drive\novell\nds\embox\edirutil.exe -?`

対話式モードで実行中に eMBox の対話型コマンドラインヘルプを表示するには、eMBox クライアントプロンプトで疑問符 (?) を入力します。たとえば、次のように入力します。  
eMBox Client> ?

ヘルプには、[486 ページの「eMBox コマンドラインクライアントのオプション」](#)で示すようなコマンドラインオプションに関する情報が表示されます。

## eMBox コマンドラインクライアントを対話式モードで実行する

対話式モードでは、eMBox コマンドを一度に1つずつ実行します。

このセクションでは、次のトピックについて説明します。

- ◆ [480 ページの「eDirectory サーバで eMBox クライアントを実行する」](#)
- ◆ [481 ページの「ワークステーションで eMBox クライアントを実行する」](#)
- ◆ [482 ページの「サーバにログインする」](#)
- ◆ [482 ページの「使用言語、タイムアウト、およびログファイルを設定する」](#)
- ◆ [483 ページの「eMTool とそのサービスを表示する」](#)
- ◆ [484 ページの「特定のサービスを実行する」](#)
- ◆ [484 ページの「現在のサーバからログアウトする」](#)
- ◆ [484 ページの「eMBox クライアントを終了する」](#)

## eDirectory サーバで eMBox クライアントを実行する

eMBox クライアントおよび Sun JVM 1.3.1 は、eDirectory と同時にインストールされています。eDirectory サーバの対話式モードで eMBox クライアントを開始するには、次の操作を行います。

- ◆ NetWare、UNIX の場合: コマンドラインから「edirutil -i」 と入力します。
- ◆ Windows の場合は、次を実行します。  
`drive\novell\nds\edirutil.exe -i`

edirutil ファイルは、eMBox クライアントを実行するためのショートカットです。このファイルには、Java 実行ファイルの場所と eMBox が eDirectory と同時にインストールされたデフォルトの場所が示されています。NetWare の場合、このファイルでは -ns オプションが指定されています (これは NetWare の Java オプションで、「新規画面」を意味します)。(Java 実行ファイルの場所は、[481 ページの「eMBox クライアント用にパスおよびクラスパスを設定する」](#)で示しているように手動で入力することもできます)。

管理対象のサーバに対して eMBox コマンドラインクライアントを使用するには、ファイアウォール越しにアクセスできる必要があります。そのため、リモートで実行する場合は、VPN アクセスが必要となります。

## ワークステーションで eMBox クライアントを実行する

eDirectory サーバではないコンピュータで eMBox クライアントを使用するには、次の操作を実行します。

- ◆ eMBoxClient.jar ファイルを eDirectory サーバから目的のコンピュータにコピーします。
  - ◆ NetWare: sys:\system\embox\eMBoxClient.jar
  - ◆ Windows: \novell\nds\embox\eMBoxClient.jar
  - ◆ UNIX: /usr/lib/nds-modules/embox/eMBoxClient.jar
- ◆ Sun JVM 1.3.1 がインストールされていることを確認します。
- ◆ 管理するサーバに対して eMBox コマンドラインクライアントを使用するためには、ファイアウォール越しにアクセスできることを確認します。

サーバの場合とは異なり、ワークステーションの場合は、edirutil コマンドをショートカットとして使用して eMBox クライアントを対話式モードで開始することはできません。パスおよびクラスパス内で環境を設定するか、パスをその都度手動で入力します。[481 ページの「eMBox クライアント用にパスおよびクラスパスを設定する」](#)を参照してください。

## eMBox クライアント用にパスおよびクラスパスを設定する

eMBox クライアントが eDirectory サーバで実行され、Java または eMBoxClient.jar ファイルの場所を変更していない場合、edirutil をショートカットとして使用して eMBox クライアントを実行できます ([480 ページの「eDirectory サーバで eMBox クライアントを実行する」](#)を参照してください)。

ただし、デフォルトの場所を変更したか、または eMBoxClient.jar ファイルをサーバでないコンピュータ上で実行している場合、あるいはクラスパスを手動で入力したい場合は、eMBox クライアントのパスおよびクラスパスをこのセクションの説明のように設定する必要があります。

次の操作を実行すれば、eMBox クライアントをコンピュータ上のどの場所からでも実行できます。

- ◆ Java 実行ファイル (Java.exe など) があるディレクトリをパスに追加するか、または Java がすでに実行されていることを確認します。

サーバでは、ほとんどの場合、すでに実行されています。Windows サーバおよび UNIX サーバでは、実行ファイルのディレクトリをパスに追加する必要があります。NetWare の場合、ディレクトリをパスに追加するのではなく、Java が実行されている必要があります。

ワークステーションの場合、手動で設定する必要がある場合があります。たとえば Windows では、[スタート] > [設定] > [コントロールパネル] > [システム] の順にクリックします。[詳細設定] タブで [環境変数] をクリックし、PATH 変数にパスを追加します。

パスを手動で入力するには、次の操作を実行します。Java 実行ファイルへのパスが追加されていない場合、eMBox を実行する前に、最初にコマンドラインで Java 実行ファイルが含まれたディレクトリへ移動する必要があります。たとえば、Windows で次のように入力します。

```
cd c:\novell\nds\embox\jre\bin
```

- ◆ eMBoxClient.jar ファイルへのパスを、クラスパスに追加します。

NetWare サーバの場合：

```
set ENVSET=path\eMBoxClient.jar
```

Windows サーバまたはワークステーションの場合：

```
set CLASSPATH=path\eMBoxClient.jar
```

UNIX サーバまたはワークステーションの場合：

```
export CLASSPATH=path\eMBoxClient.jar
```

パスを手動で入力するには、次の操作を実行します。クラスパスを指定するには、次のように eMBox を実行するたびに Java の `-cp` フラグを使用する方法もあります。

```
java -cp path/eMBoxClient.jar embox -i
```

たとえば、Windows では次のように入力します。

```
java -cp c:\novell\nds\embox\eMBoxClient.jar embox -i
```

**警告：**NetWare サーバの場合のみ、異常終了を防ぐために `-ns`(「新規画面」を開く NetWare での Java オプション)を追加します。たとえば、次のように入力します。

```
java -ns -cp sys:\system\embox\eMBoxClient.jar embox -i
```

これらの手順を実行した後は、次のコマンドを使用して、コンピュータ上のどの場所からでも対話式モードによる eMBox クライアントを実行できます。

```
java embox -i
```

**警告：**NetWare サーバの場合のみ、異常終了を防ぐために `-ns`(「新規画面」を開く NetWare での Java オプション)を追加します。たとえば、次のように入力します。

```
java -ns embox -i
```

Java コマンドについては、[Sun Web サイト \(http://java.sun.com\)](http://java.sun.com)にある Java のマニュアルを参照してください。

## サーバにログインする

サーバにログインするには、サーバ名または IP アドレス、および特定のサーバへ接続するためのポート番号を指定する必要があります。パブリックログインの場合、ユーザ名およびパスワードは必要ありません。

たとえば、eMBox クライアントを対話式モードでオープンした後で、次のように入力します。

```
login -s 137.65.123.244 -p 8008 -u admin.mycompany  
-w mypassword -n
```

ポート番号についての詳細は、[488 ページの「eDirectory ポート番号を確認する」](#)を参照してください。

## 使用言語、タイムアウト、およびログファイルを設定する

デフォルトの言語は、クライアントシステムの言語です。そのため、ほとんどの場合、特別に言語を設定する必要はありません。同様に、タイムアウトもほとんどの場合、デフォルトの設定で問題ありません。ログファイルを設定するには、ファイル名とファイルを開くモード(追加または上書き)を指定します。

次の表に、コマンド例を示します。

コマンド	説明
<code>set -L en,de</code>	使用言語を英語、ドイツ語の優先順で設定します。
<code>set -T 100</code>	タイムアウトを 100 秒に設定します。タイムアウトは、サーバからの応答を待つ時間を設定するものです。
<code>set -l mylog.txt -o</code>	mylog.txt をログファイルとして使用し、開いた後は上書きします。 デフォルトの設定は「追加」です。

## eMTool とそのサービスを表示する

サーバにログインしたら、`list` コマンドを使用して、そのサーバ上で使用できるサービスを表示できます。

`list` コマンドを使用すると、次に示す eMTool とそのサービスが動的に表示されます。

eMTool	説明
backup	Novell eDirectory バックアップ eMTool
dsmerge	Novell eDirectory マージ eMTool
dsrepair	Novell eDirectory 修復 eMTool
dsschema	Novell eDirectory スキーマ操作 eMTool
service	Novell eDirectory サービスマネージャ eMTool

リストを強制的にリフレッシュするには、`-r` を使用します。サービスの詳細を表示するには、`-t` を使用します。コマンド形式だけを表示するには、`-f` を使用します。

次の表に、コマンド例を示します。

コマンド	説明
<code>list</code>	サーバ上で使用できる eMTool を表示します。
<code>list -r</code>	eMTool リストをリフレッシュします。
<code>list -t backup</code>	backup サービスの詳細を表示します。
<code>list -t dsrepair</code>	DSRepair サービスの詳細を表示します。
<code>list -t dsmerge -f</code>	DSMerge サービスのコマンド形式だけを表示します。

## 特定のサービスを実行する

サーバにログインした後は、各 eMTool サービスを使用してタスクを実行できます。次に例を示します。

コマンド	説明
<code>dsrepair.rld</code>	ローカルデータベースを修復します。
<code>backup.getconfig</code>	バックアップ設定情報を取得します。

詳細については、次を参照してください。

- ◆ [412 ページの「eMBox クライアントを使ったバックアップ / 復元作業」](#)
- ◆ [210 ページの「eMBox クライアントを使用したツリーのマージ」](#)
- ◆ [236 ページの「eMBox クライアントを使用したデータベースの修復」](#)
- ◆ [169 ページの「eMBox クライアントのサービスマネージャ eMTool を使用する」](#)

## 現在のサーバからログアウトする

現在のセッションからログアウトするには、次のコマンドを使用します。

`logout`

別のサーバにログインする場合は、自動的に現在のサーバからログアウトされるため、このコマンドは必要ありません。

## eMBox クライアントを終了する

eMBox クライアントを終了するには、次のいずれかのコマンドを使用します。

`exit`

または

`quit`

## eMBox コマンドラインクライアントをバッチモードで実行する

eMBox クライアントをバッチモードで実行するには、次の 3 つの方法があります。

- ◆ [484 ページの「単一タスク」](#)
- ◆ [485 ページの「内部バッチファイル」](#)
- ◆ [486 ページの「システムバッチファイル」](#)

システムバッチファイルと内部バッチファイルを組み合わせて使用することで、コマンドをより自由に実行でき、頻繁に実行するコマンドの編成や再使用が可能です。

## 単一タスク

コマンドラインから単一の eMBox タスクをバッチモードで実行するには、コマンドに `-t` オプションを使用してツールとタスクを指定し、`-i` オプション（対話式モードを指定するオプション）を省くだけです。たとえば、次のように入力します。



```
java embox -s 137.65.123.244 -p 8008 -u admin.mycompany
-w mypassword -l mylog.txt -t dsrepair.rld -n
```

**警告** : NetWare の場合のみ、異常終了を防ぐために `-ns`(「新規画面」を開く NetWare での Java オプション) を追加します。たとえば、次のように入力します。

```
java -ns embox -s 137.65.123.244 -p 8008 -u admin.mycompany -w mypassword -l mylog.txt
-t dsrepair.rld -n
```

異なるサーバ上で複数のタスクを実行する場合や、頻繁に実行するタスクの場合は、内部バッチファイルを使用する方が便利です。詳細については、次のセクション「[内部バッチファイル](#)」を参照してください。

## 内部バッチファイル

**eMBox** クライアントの内部バッチファイルを使用して **eMBox** クライアントをバッチモードで実行するには、対話式モードで実行するような **eMBox** コマンドのグループを含むファイルを作成する必要があります。

**eMBox** クライアントの内部バッチファイルを使用すれば、バッチファイル内のすべてのコマンドを自動で実行できます。複数の **eMBox** ツールを使用した複数のタスクを、同一のサーバ上でタスクごとにログインとログアウトを繰り返すことなく実行できます。また 1 つのサーバから、複数のサーバに対して複数の **eMBox** ツールを使用したタスクを実行できます。

内部バッチファイルを使用すれば、頻繁に実行するコマンドの編成や再利用ができます。そのため、これらのコマンドを実行するたびに、コマンドラインから手動で入力する必要はありません。

内部バッチファイルの実行は、コマンドラインから **eMBox** クライアントコマンドを使用して行います。たとえば、次のコマンドは、サーバにログインし、`mybatch.mbx` ファイル内に列挙されたコマンドを実行します。

```
java embox -s 137.65.123.244 -p 8008 -u admin.mycompany -w mypassword -l
mylog.txt -o -b mybatch.mbx -n
```

**警告** : NetWare の場合のみ、異常終了を防ぐために `-ns`(「新規画面」を開く NetWare での Java オプション) を追加します。たとえば、次のように入力します。

```
java -ns embox -s 137.65.123.244 -p 8008 -u admin.mycompany -w mypassword -l mylog.txt
-o -b mybatch.mbx -n
```

もう 1 つの方法は、同様のコマンドをシステムバッチファイル内に記述し、そのファイルがサーバ上で自動実行されるようにスケジュールすることです。[486 ページの「システムバッチファイル」](#)を参照してください。

次に、**eMBox** 内部バッチファイルの例を示します。このファイルには、実行するコマンドの例および別のサーバへログインする例が記述されています。この例では、**eMBox** クライアントを開いたときに、サーバにログインしているものと仮定しています (それぞれのコマンドは、行を分ける必要があります。# で始まる行はコメントです)。

```
# This file is named mybatch.mbx.
# This is an example of commands you could use in
# an eMBox internal command batch file.

# Backup commands
backup.getconfig
backup.backup -b -f mybackup.bak -l backup.log -t -e -w

# DSRepair commands
dsrepair.rld
```

```

# Log in to a different server
login -s 137.65.123.255 -p 8008 -u admin.mycompany -w mypassword -n

# DSmerge commands
dsmerge.pr -u admin.mycompany -p admin.mycompany -n mypassword # Schema
Operations
dsschema.rst
dsschema.dse
dsschema.rls
dsschema.gsu
dsschema.scc
dsschema.irs -n LocalTree

# DSService commands
service.serviceList

# End of example.

```

## システムバッチファイル

他のコマンドラインツールと同様に、eMBox クライアントコマンドが含まれたシステムバッチファイルを作成し、それらをコマンドラインから手動で実行したり、サーバ上で自動で実行されるようにスケジュールしたりできます。たとえば、[415 ページの「バッチファイルと eMBox クライアントによる無人バックアップ」](#)で示す例のように、システムバッチファイルを使用して、無人バックアップを実行できます。

1つのサーバから複数のサーバに対して、複数の eMBox ツールを使用したタスクを実行できます。

システムバッチファイルでは、eMBox クライアントの単一コマンドと内部バッチファイルを組み合わせて使用できます。これにより、コマンドをより自由に実行でき、頻繁に実行するコマンドの編成や再使用ができます。詳細については、[485 ページの「内部バッチファイル」](#)を参照してください。

バッチファイルを自動で実行する方法については、ご使用のオペレーティングシステムのマニュアルまたはサードパーティ製スケジューリングソフトウェアのマニュアルを参照してください。

**注:** NetWare では、サードパーティのスケジューリングソフトウェアが使用できます。また、サポート対象のツールではありませんが、Novell Technical Support から [CRON.NLM \(http://support.novell.com/servlet/tidfinder/2939440\)](http://support.novell.com/servlet/tidfinder/2939440) をダウンロードしてお使いいただけます。

## eMBox コマンドラインクライアントのオプション

オプション	説明
-? または -h	ヘルプ情報を表示します。
-i	eMBox コマンドを一度に 1 つずつ、対話的に実行します。
-s <i>server</i>	eMBox サーバの名前または IP アドレスを指定します。 デフォルト値は 127.0.0.1 です。
-p <i>port</i>	eMBox サーバのポート番号を指定します。 デフォルト値は 80 です。

オプション	説明
-u <i>user</i>	ユーザDNを指定します。たとえば、admin.mycompanyと指定します。 デフォルト値は anonymous です。
-w <i>password</i>	-u で指定したユーザのパスワードを指定します。
-m <i>mode</i>	ログインモードを指定します。 デフォルト値は dclient です。
-n	安全な SSL 接続を試行しません。保護されていない接続を使用します。  このオプションを使用しない場合、eMBox クライアントでは SSL 接続を確立しようとします。そのため、クラスパスには JSSE ファイルが必要となり、これがない場合はエラーが返されます。詳細については、 <a href="#">487 ページの「eMBox クライアントを使用してセキュア接続を確立する」</a> を参照してください。
-l <i>log file</i>	ログファイルの名前を指定します。
-o	ログファイルを開いた後は、上書きします。
-T <i>timeout</i>	サーバから応答を待つ時間 ( 秒 ) を指定します。
-L <i>language</i>	使用できる言語を優先順にコンマ区切りで指定します。たとえば、en-US,de_DE と指定します。このオプションのデフォルトは、クライアントシステムの言語です。
-t [ <i>tool..</i> ] <i>taskoptions</i>	この接続での単一のサービスを実行します。-t に続く文字列は、有効な eMBox コマンドである必要があります。
-b <i>eMBox batch file</i>	バッチファイルで指定した一連のサービスを実行します。バッチファイル内の eMBox コマンドは、行を分けて記述する必要があります。 # で始まる行はコメントです。

## eMBox クライアントを使用してセキュア接続を確立する

非セキュア接続を使用している場合、ユーザ名やパスワードなどの入力したすべての情報は、クリアテキストでネットワーク上へ送信されます。

SSL を使用したセキュア接続を確立するには、次の操作を実行します。

- ◆ サーバへログインする際は、コマンドで -n オプションを使用しないでください。このオプションは、非セキュア接続を指定するものです。セキュア接続がデフォルトの設定です。
- ◆ クラスパスに、次に示す JSSE(Java Secure Socket Extension) ファイルがあることを確認します。
  - ◆ jsse.jar
  - ◆ jnet.jar
  - ◆ jcert.jar

これらのファイルがない場合、eMBox クライアントは、セキュア接続が確立できないことを示すエラーを返します。

これらのファイルや JSSE についての情報は、[Sun Web サイト \(http://java.sun.com/products/jsse\)](http://java.sun.com/products/jsse) から入手できます。

## eDirectory ポート番号を確認する

eMBoxクライアントでサーバにログインするには、ポート番号を指定する必要があります。eDirectoryをインストールする際にポート番号を指定した場合には、その番号を使用します。デフォルトのポート番号は、次のとおりです。

- ◆ NetWare の場合、デフォルト の非セキュアポート は 8008 で、セキュアポート は 8009 です。
- ◆ その他のプラットフォームの場合、デフォルトの非セキュアポートは8008で、セキュアポートは 8010 です。

次に示すセクションには、eDirectory に割り当てられたポートを確認するための、その他のヒントがあります。

- ◆ [488 ページの「Windows の場合」](#)
- ◆ [488 ページの「NetWare の場合」](#)
- ◆ [489 ページの「UNIX の場合」](#)

### Windows の場合

- 1 [スタート] > [設定] > [コントロールパネル] の順にクリックします。
- 2 [Novell eDirectory Services] アイコンをダブルクリックしてから、[トランスポート] タブをクリックします。
- 3 セキュアポート、または非セキュアポートを確認します。
  - ◆ 非セキュアポートを確認するには、HTTP の横のプラス記号をクリックします。
  - ◆ セキュアポートを確認するには、HTTPS の横のプラス記号をクリックします。[Bound Transports( バインドされたトランスポート )] の横のプラス記号をクリックして、ポート番号を確認します。

### NetWare の場合

サーバオブジェクトのネットワークアドレスプロパティに、ポートが表示されます。

サーバオブジェクトのネットワークアドレスプロパティは、次のツールで確認できます。

- ◆ iManager で、[eDirectory 管理] > [オブジェクトの変更] の順にクリックしてサーバオブジェクトを見つけ、[一般] タブのドロップダウンリストでネットワークアドレスを参照します。
- ◆ ConsoleOne<sup>®</sup> で、サーバオブジェクトを右クリックするかまたはサーバオブジェクトを選択した状態で [オブジェクト] > [プロパティ] の順にクリックし、ネットワークアドレスのドロップダウンリストを参照します。

始めが http: または https: で、終わりが /portal のネットワークアドレスを探します。これらのアドレスが、eMBox ツールが使用する非セキュアポートおよびセキュアポートです。

ポート番号の確認方法を次に示します。

- ◆ ネットワークアドレス内にポート番号が表示されている場合、その番号が割り当てられたポート番号です。

たとえば、http://137.65.188.1:8008/portal の場合、ポート 8008 が eMBox ツールで使用されています。

- ◆ サーバのIPアドレスだけで、ポート番号が表示されていない場合、デフォルトのポート番号が使用されています。

たとえば、`https://137.65.188.1/portal` の場合、IP アドレスの後にはポート番号が表示されていませんが、これは eMBox ツールでデフォルトのポート番号が使用されていることを示しています。デフォルトのポート番号は、NetWare では 8009、それ以外のプラットフォームでは 8010 です。

## UNIX の場合

次のコマンドを使用して、ポートを表示できます。

```
ndsconfig get | grep http
```

`http.server.interface`、その次にポート番号という表現で記述された行を探します。

また、[488 ページの「NetWare の場合」](#)で説明している方法を使用し、iManager または ConsoleOne によりポート番号を確認することもできます。

## eMBox ログの記録の使用

eMBox ログの記録はインフラストラクチャモジュールで、DSBackup、DSMerge、DSRepair などの、すべての eMTool(eDirectory Management Tool) のイベントを記録します。このリリースで提供されているログファイルは 1 つだけです。このログファイルに、すべての eMTool の操作が記録されます。

eMBox ログの記録は、クライアントログサービスとは異なります。クライアントログサービスは、eMBox クライアントを実行する際に、ログファイルを指定することで提供されるものです(たとえば、eMBox クライアントコマンドで「-l mylogfile.txt」と指定する場合、または iManager でログファイル名として「mylogfile.txt」を入力する場合です)。現在、eMBox ログの記録は eMBox が実行するタスクに対するすべてのサーバメッセージを記録し、その内容は詳細なものです。これに対して、クライアントログサービスではクライアントメッセージおよびクライアントに送信されたメッセージが記録され、進捗状況の概要がレポートされます。

ログの記録は非同期で実行され、デフォルトではすべての操作が記録されます。

このリリースの eMBox ログの記録には、次の機能があります。

- ◆ ログファイルの名前と場所を変更できます。  
デフォルトでは、ログファイルは eDirectory がインストールされたディレクトリ内の `emboxlog` ディレクトリに作成されます。
- ◆ 最大ファイルサイズが変更できます。変更後、ログファイルはリセットされます。  
最大ファイルサイズは 8MB です。
- ◆ ログモードを変更できます。  
すべての新しいメッセージをログファイルに追加するか、または既存のログファイルを上書きするかを選択できます。デフォルトでは、追加のオプションが指定されています。
- ◆ ログの記録を開始または停止できます。  
デフォルトでは、eMBox が起動するとログの記録は開始モードになります。停止モードの間は、メッセージは記録されません。
- ◆ ログファイルの内容をリセットできます。
- ◆ クライアントコンピュータからログファイルを読み込むことができます。

このセクションでは、次のトピックについて説明します。


- ◆ 490 ページの「[eMBox ログの記録コマンドラインクライアントを使用する](#)」
- ◆ 490 ページの「[Novell iManager で eMBox ログの記録機能を使用する](#)」

## eMBox ログの記録コマンドラインクライアントを使用する

次の表に、eMBox ログの記録コマンドラインクライアントのオプションを示します。

オプション	説明
logstart	eMBox ログの記録を開始します。
logstop	eMBox ログの記録を停止します。
readlog	現在のログファイルを表示します。
getlogstate	現在の eMBox ログの記録の状態 ( 開始または停止 ) を表示します。
getloginfo	eMBox ログファイルの名前、ログモード ( 追加または上書き )、最大サイズと現在のサイズを表示します。
setloginfo [-f <i>filename</i> ] [-s <i>size in Kilo bytes</i> ] [-a   -o]	次のパラメータを使用して、eMBox ログファイルの名前、サイズ、ログモード ( 追加または上書き ) を設定します。 <ul style="list-style-type: none"><li>◆ -f <i>filename</i> eMBox ログファイルの名前を設定します。</li><li>◆ -s <i>size in KB</i> ログファイルの最大サイズを設定します。</li><li>◆ -a 新しいログメッセージを現在のログメッセージに追加します。</li><li>◆ -o ログファイルを上書きします。</li></ul>
emptylog	サーバログファイルの内容をクリアします。

## Novell iManager で eMBox ログの記録機能を使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] ユーティリティ > [ログファイル] の順にクリックします。
- 3 ログファイル操作を実行するサーバを指定してから、[次へ] をクリックします。
- 4 サーバへの認証を行ってから、[次へ] をクリックします。
- 5 実行するログファイル操作を選択します。

詳細については、[ヘルプ] をクリックしてください

# 18

## Novell eDirectory のトラブルシューティング

次のセクションでは、Novell® eDirectory™ ソフトウェアに関する問題を解決するための提案およびリソースについて説明します。

- ◆ 491 ページの「エラーコードの解決」
- ◆ 491 ページの「オブジェクトの命名」
- ◆ 492 ページの「eDirectory バージョン番号の確認」
- ◆ 496 ページの「eDirectory レプリカの問題から回復する」
- ◆ 496 ページの「eDirectory for Windows サーバが起動しない場合」
- ◆ 497 ページの「Windows サーバが eDirectory データベースファイルを開けない場合」
- ◆ 497 ページの「緊急修復後の Windows での eDirectory の復元」
- ◆ 498 ページの「ログファイル」
- ◆ 498 ページの「LDIF ファイルのトラブルシューティング」
- ◆ 516 ページの「Linux、Solaris、AIX、および HP-UX での eDirectory のトラブルシューティング」
- ◆ 536 ページの「ConsoleOne のトラブルシューティング」
- ◆ 537 ページの「破損通知」
- ◆ 543 ページの「DS がロードされない場合の HTTPSTK へのアクセス」
- ◆ 545 ページの「SNMP のトラブルシューティング」
- ◆ 547 ページの「コンテナ管理者の eDirectory インストールの失敗」
- ◆ 547 ページの「Sun One スキーマの Novell eDirectory への移行」
- ◆ 550 ページの「ICE を使用した、アクティブディレクトリスキーマの Novell eDirectory への移行」

### エラーコードの解決

eDirectory エラーコードの完全なリストおよび説明については、[Novell Error Codes Web ページ \(http://www.novell.com/documentation/japanese/nwec/index.html\)](http://www.novell.com/documentation/japanese/nwec/index.html) を参照してください。

### オブジェクトの命名

オブジェクト名に特殊文字を使用すると、-671ペアレントがありませんというエラーメッセージが表示されます。オブジェクト名には、次の特殊文字を使用しないでください。

\\, \* ? .

## eDirectory バージョン番号の確認

次のセクションでは、サーバにインストールされている eDirectory のバージョンを確認する方法を示します。

- ◆ 492 ページの「NetWare の場合」
- ◆ 493 ページの「Windows の場合」
- ◆ 493 ページの「Linux の場合」
- ◆ 494 ページの「Solaris の場合」
- ◆ 495 ページの「AIX の場合」
- ◆ 495 ページの「HP-UX の場合」

### NetWare の場合

- ◆ ds.nlm またはその他の .nlm を実行する。  
サーバコンソールに、**ds.nlm** と入力します。これにより、マーケティング文字列 (Novell eDirectory 8.7 など) および内部ビルド番号 (DS v10410.xx など) が表示されます。
- ◆ iMonitor を実行する。  
[エージェントの概要] ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。  
iMonitor を実行する方法の詳細については、173 ページの「iMonitor へアクセスする」を参照してください。
- ◆ nwconfig を実行する。  
サーバコンソールで「**nwconfig**」と入力して、[製品情報] を選択します。インストールで登録された eDirectory (NDS® eDirectory 8.5 以降) の場合、インストール実行時に登録された eDirectory の情報が表示されます。通常は、マーケティング番号およびビルド番号が一緒に表示されます。たとえば、マーケティングバージョン eDir 8.5 およびビルドバージョン 85.01 などのように表示されます。
- ◆ LDAP に DStTrace を通して環境設定を表示する。  
これは、ロードを実行するほとんどのユーティリティ (DSRepair または DSMerge など) においても同様です。この方法で内部ビルド番号が表示されます。  
DStTrace の詳細については、『Looking Into the Directory Services Trace (DStTrace) Options (<http://developer.novell.com/research/sections/netmanage/dirprimer/2001/august/spv.htm>)』および『More on Using the DStTrace Command (<http://developer.novell.com/research/sections/netmanage/dirprimer/2001/septembe/p010901.htm>)』を参照してください。
- ◆ eDirectory のダウンロードファイル名を読み込む。  
eDirectory のダウンロードファイル名は、通常、マーケティング文字列と一致します。たとえば、Novell eDirectory 8.7 のダウンロードファイル名は edir871.exe です。
- ◆ コンソールにプロンプトが表示されたら、「**version**」と入力します。  
これにより、eDirectory のバージョンが表示されます。



## Windows の場合

- ◆ iMonitor を実行する。

[エージェントの概要] ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。

iMonitor を実行する方法の詳細については、[173 ページの「iMonitor へアクセスする」](#)を参照してください。

- ◆ NDSCons.exe を実行する。

Windows の [コントロールパネル] で、[Novell eDirectory Services] をダブルクリックします。[サービス] カラムで、`ds.dlm` を選択し、[設定] をクリックします。[エージェント] タブに、マーケティング文字列 (Novell eDirectory 8.7.1 など) および内部ビルド番号 (10510.64 など) が表示されます。

- ◆ eDirectory ユーティリティを実行する。

ほとんどの eDirectory ユーティリティの [ヘルプ] メニューには [バージョン情報] オプションがあり、該当するユーティリティのバージョン番号 (Merge Graft Utility 10510.35 など) が表示されます。内部ビルド番号が、ユーティリティのメインラベル (DSRepair - Version 10510.37 など) に表示される場合もあります。

eDirectory ユーティリティ (DSMerge または DSRepair など) をロードするには、Windows の [コントロールパネル] で [Novell eDirectory Services] をダブルクリックします。[サービス] カラムで、ユーティリティを選択し、[開始] をクリックします。

- ◆ eDirectory の .dlm ファイルのプロパティを表示する。

Windows Explorer の .dlm ファイルを右クリックし、[プロパティ] ダイアログボックスの [バージョン] タブをクリックします。これにより、ユーティリティのバージョン番号が表示されます。eDirectory の .dlm ファイルがデフォルトで格納される場所は、`C:\novell\NDS` です。

## Linux の場合

- ◆ `ndsstat` を実行する。

`ndsstat` ユーティリティには、eDirectory ツリー名、完全識別サーバ名、および eDirectory バージョンなどの eDirectory サーバに関する情報が表示されます。次の例では、eDirectory 8.7.1 が製品バージョン (マーケティング文字列) で、10510.65 がバイナリバージョン (内部ビルド番号) です。

```
osg-dt-srv17: />ndsstat
Tree Name:SNMP-HPUX-RASH
Server Name: .CN=osg-dt-srv17.O=novell.T=SNMP-HPUX-RASH.
Binary Version:10510.65
Root Most Entry Depth:0
Product Version:NDS/Unix - NDS eDirectory v8.7.1 [DS]
```

`ndsstat` の実行の詳細については、[559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」](#) または `ndsstat` マニュアルページ (`ndsstat.1m`) を参照してください。

- ◆ `ndsd --version` を実行する。

`ndsd` の実行の詳細については、[559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」](#) または `ndsd` マニュアルページ (`ndsd.1m`) を参照してください。

- ◆ iMonitor を実行する。

[エージェントの概要] ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。

iMonitor を実行する方法の詳細については、173 ページの「iMonitor へアクセスする」を参照してください。

- ◆ rpm -qi NDSserv を実行する。

このコマンドを入力すると、ndsd --version に似た情報が表示されます。

## Solaris の場合

- ◆ ndsstat を実行する。

ndsstat ユーティリティには、eDirectory ツリー名、完全識別サーバ名、および eDirectory バージョンなどの eDirectory サーバに関する情報が表示されます。次の例では、eDirectory 8.7.1 が製品バージョン (マーケティング文字列) で、10510.65 がバイナリバージョン (内部ビルド番号) です。

```
osg-dt-srv17: />ndsstat
Tree Name:SNMP-HPUX-RASH
Server Name:.CN=osg-dt-srv17.O=novell.T=SNMP-HPUX-RASH.
Binary Version:10510.65
Root Most Entry Depth:0
Product Version:NDS/Unix - NDS eDirectory v8.7.1 [DS]
```

ndsstat の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsstat マニュアルページ (ndsstat.1m) を参照してください。

- ◆ ndsd --version を実行する。

ndsd の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsd マニュアルページ (ndsd.1m) を参照してください。

- ◆ iMonitor を実行する。

エージェントの概要ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。

iMonitor を実行する方法の詳細については、173 ページの「iMonitor へアクセスする」を参照してください。

- ◆ pkginfo -l NDSserv を実行する。

このコマンドを入力すると、ndsd --version に似た情報が表示されます。

## AIX の場合

- ◆ ndsstat を実行する。

ndsstat ユーティリティには、eDirectory ツリー名、完全識別サーバ名、および eDirectory バージョンなどの eDirectory サーバに関する情報が表示されます。次の例では、eDirectory 8.7.1 が製品バージョン (マーケティング文字列) で、10510.65 がバイナリバージョン (内部ビルド番号) です。

```
osg-dt-srv17:/>ndsstat
Tree Name:SNMP-HPUX-RASH
Server Name:.CN=osg-dt-srv17.O=novell.T=SNMP-HPUX-RASH.
Binary Version:10510.65
Root Most Entry Depth:0
Product Version:NDS/Unix - NDS eDirectory v8.7.1 [DS]
```

ndsstat の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsstat マニュアルページ (ndsstat.1m) を参照してください。

- ◆ ndsd --version を実行する。

ndsd の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsd マニュアルページ (ndsd.1m) を参照してください。

- ◆ iMonitor を実行する。

[エージェントの概要] ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。

iMonitor を実行する方法の詳細については、173 ページの「iMonitor ヘアアクセスする」を参照してください。

## HP-UX の場合

- ◆ ndsstat を実行する。

ndsstat ユーティリティには、eDirectory ツリー名、完全識別サーバ名、および eDirectory バージョンなどの eDirectory サーバに関する情報が表示されます。次の例では、eDirectory 8.7.1 が製品バージョン (マーケティング文字列) で、10510.65 がバイナリバージョン (内部ビルド番号) です。

```
osg-dt-srv17:/>ndsstat
Tree Name:SNMP-HPUX-RASH
Server Name:.CN=osg-dt-srv17.O=novell.T=SNMP-HPUX-RASH.
Binary Version:10510.65
Root Most Entry Depth:0
Product Version:NDS/Unix - NDS eDirectory v8.7.1 [DS]
```

ndsstat の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsstat マニュアルページ (ndsstat.1m) を参照してください。

- ◆ ndsd --version を実行する。

ndsd の実行の詳細については、559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」または ndsd マニュアルページ (ndsd.1m) を参照してください。

- ◆ iMonitor を実行する。

[エージェントの概要] ページで [認識サーバ] をクリックします。次に、[データベースで認識されているサーバ] の下にある [認識サーバ] をクリックします。[エージェントリビジョン] カラムに各サーバの内部ビルド番号が表示されます。たとえば、Novell eDirectory 8.7.1 のエージェントリビジョン番号は 10510.64 などです。

iMonitor を実行する方法の詳細については、[173 ページの「iMonitor へアクセスする」](#)を参照してください。

## eDirectory レプリカの問題から回復する

eDirectory では、Novell の強力なディレクトリサービス、および複製による障害対策が提供されています。複製により、eDirectory データベースまたはその一部のコピーを、複数のサーバで同時に保持できます。

eDirectory パーティションのレプリカは、常に複数作成しておいてください。レプリカを複数作成しておくことで、あるレプリカがハードディスクの故障で破損したり失われたりした場合でも、ConsoleOne<sup>®</sup> または Novell iManager を使用してそのレプリカを削除し、破損していないレプリカから作成した新しいレプリカに置き換えることができます。

レプリカの削除の詳細については、[122 ページの「レプリカを削除する」](#)を参照してください。

## eDirectory for Windows サーバが起動しない場合

Windows サーバのブート時に eDirectory サーバが起動に失敗すると、サービスの開始に失敗したことを示すメッセージが通知されます。

他に eDirectory データベースのレプリカがない場合、ユーザはログインできません。

他にレプリカがある場合は、ログインが遅くなります。また、これらのレプリカを保持しているサーバ上に通信エラーおよび同期エラーが表示されます。

- ◆ Windows レジストリ内の eDirectory サーバエントリが編集されたか、Windows レジストリが破損している可能性があります。
- ◆ eDirectory データベースファイルが破損しているか、削除された可能性があります。
- ◆ 他のサービスが開始されていないために eDirectory サーバが起動できない場合、[スタート] > [プログラム] > [管理ツール] > [イベントビューア] の順にクリックして詳細を参照できます。

eDirectory サーバを起動する前に、他の必要なサービスにおける問題を解決する必要があります。

- ◆ レジストリまたは eDirectory 実行可能ファイルが、破損しているか失われています。システムディレクトリにある SAMMIG ユーティリティを実行します。[Windows NT 上の NDS のアンインストール] を選択し、NT ドメインに新しい eDirectory 情報を追加します。操作を続行してアンインストールを完了します。次に、sammig.exe を再起動し、eDirectory のインストールへ進みます。
- ◆ データベースファイルが、破損しているか削除されています。NT サーバ上で eDirectory サーバが起動しているのに、サービスが eDirectory データベースファイルを開くことができない場合は、[497 ページの「Windows サーバが eDirectory データベースファイルを開けない場合」](#)を参照してください。
- ◆ eDirectory サーバがハブまたはスイッチに接続されていないか、クロスケーブルを使用してワークステーションに直接接続されていません。サーバをハブまたはスイッチに接続してください。

## Windows サーバが eDirectory データベースファイルを開けない場合

eDirectory サーバがデータベースファイルを開けない場合、Windows サーバ上で、これを示すメッセージが通知されます。

他にデータベースのレプリカがない場合、ユーザはログインできません。

他にレプリカがある場合は、ログインが遅くなります。また、これらのレプリカを保持しているサーバ上に通信エラーおよび同期エラーが表示されます。

- ◆ NT/2000 サーバ上のディスクエラーのため、データベースファイルが破損している可能性があります。
- ◆ 他のユーザにより、1つまたは複数のデータベースファイルが削除された可能性があります。

他に eDirectory データベースのレプリカがある場合は、次の手順を実行します。

- 1 管理ワークステーションから Novell iManager を起動します。
- 2 破損したレプリカをレプリカリングから削除します。

詳細については、[122 ページの「レプリカを削除する」](#)を参照してください。

- 3 sammig.exe ユーティリティを NT サーバ上のシステムディレクトリ (通常は、c:\winnt\system32) から実行するか、[スタート] メニュー ( [スタート] > [プログラム] > [管理ツール (共通)] > [Migration Tool for NetWare] ) から実行します。
- 4 eDirectory サーバ上に新しいレプリカを作成するオプションを選択します。

この eDirectory サーバだけにパーティションのレプリカがある場合は、次の手順を実行します。

- 1 sammig.exe ユーティリティを NT サーバ上のシステムディレクトリ (通常は、c:\winnt\system32) から実行するか、[スタート] メニュー ( [スタート] > [プログラム] > [管理ツール (共通)] > [Migration Tool for NetWare] ) から実行します。
- 2 [Windows 上の NDS のアンインストール] を選択して、移行前の Windows ドメインの状態に戻します。
- 3 操作を続行してアンインストールを完了します。
- 4 Migration Tool for NetWare を再起動し、Windows に eDirectory をインストールします。
- 5 ユーザオブジェクトを NT/2000 ドメインから eDirectory ツリーへ移動します。

## 緊急修復後の Windows での eDirectory の復元

Windows サーバ上で緊急修復をする以外に選択の余地がない場合があります。このとき、緊急修復ディスクがない場合や、緊急修復ディスクが eDirectory のインストール前に作成されたものである場合、eDirectory クライアントとレジストリ設定は削除されます。nds4nter.exe ユーティリティで必要なレジストリ設定を復元して、eDirectory ファイルを再ロードできます。

\\i386\goodies ディレクトリから、nds4nter.exe を実行します。

緊急修復が完了したら、CD から緊急修復ユーティリティを実行します。ユーティリティは最初にレジストリ設定をいくつか復元し、それから eDirectory のインストールを起動します。インストールプログラムによってファイルがコピーされたら、再起動オプションを選択する必要があります。再起動後、ユーザは移行されたドメインにアクセスできます。

## ログファイル

このセクションには、次のログファイルに関する情報が含まれています。

- ◆ 498 ページの「[modschema.log](#)」
- ◆ 498 ページの「[dsinstall.log](#)」

### modschema.log

modschema.log ファイルには、eDirectory サーバを既存のツリーにインストールするときに適用されるすべてのスキーマ拡張の結果が含まれています。ログの各行は、どのクラスまたは属性が追加または変更されたか、および更新の試行のステータスを示します。

このログは、インストール処理を実行するたびに、作成されるか上書きされます。したがって、このログは、最後に行われた試行の結果だけを表します。このログには、eDirectory スキーマ拡張に加えて、新しい eDirectory サーバを追加する前に DSINSTALL フロントエンドによって適用された他のスキーマ拡張 (LDAP、SAS など) の結果も含まれます。

スタンドアロンサーバをインストールする場合またはターゲットサーバの eDirectory バージョンが 7.01 以降である場合は、このログは生成されません。

### dsinstall.log

ログの最初の部分は、設定されている環境変数を表示します。2番目の部分には、eDirectory インストール処理を記録するステータスメッセージが含まれています。

## LDIF ファイルのトラブルシューティング

Novell インポート / エクスポート変換ユーティリティを使用すると、eDirectory との間での LDIF ファイルのインポートおよびエクスポートが簡単になります。詳細については、[131 ページの「Novell インポート / エクスポート変換ユーティリティ」](#)を参照してください。

LDIF インポートを正しく機能させるには、Novell インポート / エクスポート変換ユーティリティが読み込み、および処理できる LDIF ファイルを最初に作成する必要があります。このセクションでは、LDIF ファイル形式および構文について説明し、正しい LDIF ファイルの例を示します。

### LDIF について

LDIF は、広く一般的に使用されているファイル形式で、ディレクトリ情報およびディレクトリで実行可能な変更操作について記述します。LDIF は、実際のディレクトリ内で使用されている記憶フォーマットとは完全に独立していて、通常は、LDAP サーバとの間でディレクトリ情報をエクスポートまたはインポートするために使用します。

一般的に、LDIF は簡単に生成できます。そのため、awk や perl などのツールを使用して、固有の形式のデータを LDAP ディレクトリに移動できます。また、LDIF 形式でテストデータを生成するスクリプトを作成することもできます。

## LDIF ファイル形式

Novell インポート/エクスポート変換ユーティリティを使用してインポートするファイルの形式は、LDIF 1 である必要があります。次に LDIF 1 形式のファイルの基本ルールを示します。

- ◆ コメント行以外の第 1 行目には、「version: 1」と記述します。
- ◆ バージョンの指定の後に、1 つ以上のレコードを定義します。
- ◆ 各レコードは、フィールドで構成されます。1 行に 1 フィールドずつ指定します。
- ◆ 各行は、改行またはキャリッジリターンと改行の組み合わせのどちらかで区切られます。
- ◆ レコードは、1 行以上の空白行で区切られます。
- ◆ LDIF レコードには、内容レコードと変更レコードの 2 つのタイプがあります。LDIF ファイルに記述するレコード数に制限はありませんが、記述されたすべてのレコードのタイプが一致している必要があります。同じ LDIF ファイル内に、内容レコードと変更レコードの両方を記述することはできません。
- ◆ シャープ記号 (#) で始まる行はコメント行です。この行は、LDIF ファイルの処理時には無視されます。

## LDIF 内容レコード

LDIF 内容レコードは、エントリ全体の内容を表します。次に、4 つの内容レコードが定義された LDIF ファイルの例を示します。

```
1 version: 1
2 dn: c=US
3 objectClass: top
4 objectClass: country
5
6 dn: l=San Francisco, c=US
7 objectClass: top
8 objectClass: locality
9 st: San Francisco
10
11 dn: ou=Artists, l=San Francisco, c=US
12 objectClass: top
13 objectClass: organizationalUnit
14 telephoneNumber: +1 415 555 0000
15
16 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
17 sn: Michaels
18 givenname: Peter
19 objectClass: top
20 objectClass: person
21 objectClass: organizationalPerson
22 objectClass: inetOrgPerson
23 telephonenumber: +1 415 555 0001
24 mail: Peter.Michaels@aaa.com
25 userpassword: Peter123
26
```

この LDIF ファイルは、次の部分から構成されています。

コンポーネント	説明
バージョン指定子	<p>LDIF ファイルの第 1 行目にはバージョンが記述されます。コロンとバージョン番号 (現在の定義は 1) の間には、1 つ以上のスペースを指定できますが、スペースを指定しなくても問題はありません。</p> <p>バージョンを指定した行がない場合、LDIF ファイルを処理するアプリケーションはそのファイルのバージョンを 0 とみなすことができます。また、構文エラーとして LDIF ファイルが拒否される可能性があります。LDIF を処理する Novell のユーティリティは、バージョンを指定する行がない場合、ファイルのバージョンを 0 とみなします。</p>
識別名指定子	<p>各内容レコードの先頭の行 (この例では、2、6、11、および 16 行目) には、そのレコードが表すエントリの DN (識別名) を指定します。</p> <p>DN 指定子は、次の 2 つのどちらかの形式をとる必要があります。</p> <ul style="list-style-type: none"> <li>◆ dn:safe_UTF-8_distinguished_name</li> <li>◆ dn::Base64_encoded_distinguished_name</li> </ul>
行区切り記号	<p>行区切り記号としては、改行、またはキャリッジリターンと改行の組み合わせのどちらかを使用できます。これにより、行区切りとして改行を使用する Linux および Solaris テキストファイルと、キャリッジリターンと改行の組み合わせを使用する MS-DOS* および Windows テキストファイルとの間の非互換性を解決できます。</p>
レコード区切り記号	<p>レコード区切りとしては、空白行 (この例では 5、10、15 および 26 行目) を使用します。</p> <p>LDIF ファイル内の各レコード (最後のレコードも含む) の終わりには、レコード区切り記号として 1 行以上の空白行を挿入する必要があります。一部のアプリケーションでは、レコード区切りを指定していない LDIF ファイルもそのまま受け入れられますが、LDIF の仕様ではレコード区切りは必須です。</p>
属性値指定子	<p>内容レコード内のその他すべての行は、値指定子です。値指定子は、次の 3 つの形式のいずれかをとる必要があります。</p> <ul style="list-style-type: none"> <li>◆ 属性の記述 :value</li> <li>◆ 属性の記述 ::Base64_encoded_value</li> <li>◆ 属性の記述 :&lt; URL</li> </ul>

## LDIF 変更レコード

LDIF 変更レコードには、ディレクトリに加えられる変更が記述されます。LDAP の更新操作 (追加、削除、変更、および DN の変更) はすべて、LDIF 変更レコードに記述できます。

LDIF 変更レコードでは、LDIF 内容レコードと同じ形式の識別名指定子、属性値指定子、およびレコード区切り記号を使用します (詳細については、[499 ページの「LDIF 内容レコード」](#)を参照してください)。LDIF 内容レコードとの違いは、LDIF 変更レコードには changetype フィールドがあることです。changetype フィールドは、変更レコードが指定する操作を識別します。

changetype フィールドは、次の 5 つの形式のいずれかをとる必要があります。



形式	説明
changetype:add	この変更レコードで LDAP の追加操作が指定されていることを示すキーワードです。
changetype:delete	この変更レコードで LDAP の削除操作が指定されていることを示すキーワードです。
changetype:moddn	この変更レコードで、LDIF プロセッサがバージョン 3 クライアントとして LDAP サーバにバインドされている場合は LDAP の DN 変更操作が、バージョン 2 クライアントとして LDAP サーバにバインドされている場合は RDN 変更操作が指定されていることを示すキーワードです。
changetype:modrdn	moddn 変更タイプと同義です。
changetype:modify	この変更レコードで LDAP の変更操作が指定されていることを示すキーワードです。

### 「追加」変更タイプ

追加変更レコードは、内容変更レコード (499 ページの「LDIF 内容レコード」を参照) に、`changetype: add` フィールドを属性値フィールドの直前に追加したものと同じです。

すべてのレコードのタイプが一致している必要があります。内容レコードと変更レコードを同じファイルに記述することはできません。

```

1 version: 1
2 dn: c=US
3 changetype: add
4 objectClass: top
5 objectClass: country
6
7 dn: l=San Francisco, c=US
8 changetype: add
9 objectClass: top
10 objectClass: locality
11 st: San Francisco
12
14 dn: ou=Artists, l=San Francisco, c=US
15 changetype: add
16 objectClass: top
17 objectClass: organizationalUnit
18 telephoneNumber: +1 415 555 0000
19
20 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
21 changetype: add
22 sn: Michaels
23 givenname: Peter
24 objectClass: top
25 objectClass: person
26 objectClass: organizationalPerson
27 objectClass: iNetOrgPerson
28 telephonenumber: +1 415 555 0001
29 mail: Peter.Michaels@aaa.com
30 userpassword: Peter123
31

```

## 「削除」変更タイプ

削除変更レコードはエントリの削除を指定するので、削除変更レコードに必要なフィールドは識別名指定子と「削除」変更タイプだけです。

次に、501 ページの「追加」変更タイプ」の LDIF ファイルで作成した 4 つのエントリを削除する LDIF ファイルの例を示します。

**重要**：以前に追加したエントリを削除するには、エントリの指定順序を逆にする必要があります。順序を逆にしないと、コンテナ内のエントリが空でないため削除操作が失敗します。

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 changetype: delete
4
5 dn: ou=Artists, l=San Francisco, c=US
8 changetype: delete
9
10 dn: l=San Francisco, c=US
11 changetype: delete
12
13 dn: c=US
14 changetype: delete
15
```

## 「変更」変更タイプ

「変更」変更タイプでは、すでに存在するエントリに対して属性値の追加、削除、および置換を指定できます。変更指定子は、次の 3 つの形式のいずれかをとる必要があります。

エレメント	説明
add: 属性タイプ	この属性タイプに対する後続の属性値指定子がエントリに追加されるように指定する必要があることを示すキーワードです。
delete: 属性タイプ	この属性タイプの値が削除されることを示すキーワードです。delete フィールドの後に属性値指定子が続く場合は、その指定された値が削除されます。  delete フィールドの後に属性値指定子がない場合は、すべての値が削除されます。属性に値がない場合、この操作は失敗しますが、属性には削除する値がないので結果的にはこの操作が成功したときと同じです。
replace: 属性タイプ	属性タイプの値が置き換えられることを示すキーワードです。replace フィールドに続く属性値指定子が、その属性タイプの新しい値になります。  replace フィールドの後に属性値指定子がない場合は、現在の値のセットが空の値のセットに置き換えられます (結果的に、属性が削除されます)。delete 変更指定子とは異なり、属性に値が設定されていない場合でもreplaceは成功します。どちらの場合も実際に得られる結果は同じです。

次の「変更」変更タイプの例では、cn=Peter Michaels エントリに別の電話番号を追加します。

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 changetype: modify
4 # add the telephone number to cn=Peter Michaels
4 add: telephonenumber
5 telephonenumber: +1 415 555 0002
6
```

1 つの LDAP 変更要求にさまざまな変更を組み合わせて指定できるのと同じように、1 つの LDIF レコードに複数の変更指定子を指定できます。ハイフン (-) だけが記述されている行は、各変更指定子に対する属性値指定の終わりを示します。

次の LDIF ファイルの例では、複数の変更を組み合わせて指定しています。

```
1 version: 1
2
3 # An empty line to demonstrate that one or more
4 # line separators between the version identifier
5 # and the first record is legal.
6
7 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
8 changetype: modify
9 # Add an additional telephone number value.
10 add: telephonenumber
11 telephonenumber: +1 415 555 0002
12 -
13 # Delete the entire facsimiletelephonenumber attribute.
14 delete: facsimileTelephoneNumber
15 -
16 # Replace the existing description (if any exists)
17 # with two new values.
18 replace: description
19 description: guitar player
20 description: solo performer
21 -
22 # Delete a specific value from the telephonenumber
23 # attribute.
24 delete: telephonenumber
25 telephonenumber: +1 415 555 0001
26 -
27 # Replace the existing title attribute with an empty
28 # set of values, thereby causing the title attribute to
29 # be removed.
30 replace: title
31 -
32
```

## 「DN 変更」変更タイプ

「DN 変更」変更タイプでは、エントリのリネーム、移動、またはその両方ができます。この変更タイプは、2つの必須フィールドと1つのオプションフィールドで構成されます。

フィールド	説明
newrdn (必須)	<p>このレコードの処理の実行中にエントリに割り当てられる新しい名前を指定します。新規 RDN(newrdn) 指定子は、次の2つのどちらかの形式をとる必要があります。</p> <ul style="list-style-type: none"><li>◆ newrdn:safe_UTF-8_relative_distinguished_name</li><li>◆ newrdn::Base64_encoded_relative_distinguished_name</li></ul> <p>新規 RDN 指定子は、「DN 変更」変更タイプが指定されたすべての LDIF レコードで指定されている必要があります。</p>
deleteoldrdn (必須)	<p>旧 RDN 削除 (deleteoldrdn) 指定子は、古い RDN を newrdn (新規 RDN) に置き換えるか、残しておくかを指定するフラグです。これは、次の2つのどちらかの形式をとります。</p> <ul style="list-style-type: none"><li>◆ deleteoldrdn: 0 リネーム後も古い RDN の値をエントリ内に残しておくことを指定します。</li><li>◆ deleteoldrdn: 1 エントリのリネーム後に古い RDN の値を削除することを指定します。</li></ul>
newsuperior (オプション)	<p>新規スーパーリア (newsuperior) 指定子は、この DN 変更レコードの処理時にエントリに割り当てる新しいペアレントの名前を指定します。新規スーパーリア指定子は、次の2つのどちらかの形式をとります。</p> <ul style="list-style-type: none"><li>◆ newsuperior:safe_UTF-8_distinguished_name</li><li>◆ newsuperior::Base64_encoded_distinguished_name</li></ul> <p>新規スーパーリア指定子は、「DN 変更」変更タイプが指定された LDIF レコードでオプションとして使用できます。これは、エントリのペアレントを変更する場合のみ指定します。</p>

次の「DN 変更」変更タイプの例で、エントリの名前を変更する方法を示します。

```
1 version: 1
2
3 # Rename ou=Artists to ou=West Coast Artists, and leave
4 # its old RDN value.
5 dn: ou=Artists,l=San Francisco,c=US
6 changetype: moddn
7 newrdn: ou=West Coast Artists
8 deleteoldrdn: 1
9
```

次の「DN 変更」変更タイプの例で、エントリを移動する方法を示します。

```
1 version: 1
2
3 # Move cn=Peter Michaels from
4 # ou=Artists,l=San Francisco,c=US to
5 # ou=Promotion,l=New York,c=US and delete the old RDN.
```

```
5 dn: cn=Peter Michaels,ou=Artists,l=San Francisco,c=US
6 changetype: moddn
7 newrdn: cn=Peter Michaels
8 deleteoldrdn: 1
9 newsuperior: ou=Promotion,l=New York,c=US
10
```

次の「DN 変更」変更タイプの例では、エントリを移動し、同時に名前を変更する方法を示します。

```
1 version: 1
2
3 # Move ou=Promotion from l=New York,c=US to
4 # l=San Francisco,c=US and rename it to
5 # ou=National Promotion.
6 dn: ou=Promotion,l=New York,c=US
7 changetype: moddn
8 newrdn: ou=National Promotion
9 deleteoldrdn: 1
10 newsuperior: l=San Francisco,c=US
11
```

**重要:** LDAP 2 の RDN 変更操作では、エントリの移動はサポートされません。LDAP 2 クライアントで LDIF newsuperior 構文を使用してエントリを移動しようとすると、その要求は失敗します。

## LDIF ファイル内での行の折り返し

LDIF ファイル内で行を折り返すには、行を折り返したい場所で単に行区切り記号 ( 改行、またはキャリッジリターンと改行の組み合わせ ) を挿入し、その後にスペースを追加します。行の先頭にスペースがある場合、LDIF パーサではスペースの後のデータとその前の行のデータを結合して解析します。したがって、先頭のスペースは無視されます。

マルチバイトの UTF-8 文字の途中では、行を折り返さないでください。

次に、行の折り返しを含む (13 および 14 行目) LDIF ファイルの例を示します。

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 sn: Michaels
4 givenname: Peter
5 objectClass: top
6 objectClass: person
7 objectClass: organizationalPerson
8 objectClass: inetOrgPerson
9 telephonenumber: +1 415 555 0001
10 mail: Peter.Michaels@aaa.com
11 userpassword: Peter123
12 description: Peter is one of the most popular music
13 ians recording on our label.He's a big concert dr
14 aw, and his fans adore him.
15
```

## LDIF ファイル内でのハッシュ化パスワードの表記

LDIF ファイル内では、ハッシュ化パスワードは Base64 データとして表記されます。属性名 *userpassword* に続けて、パスワードをハッシュ化するために使用される暗号化方式の名前を記述する必要があります。この名前は、次に示すように中カッコ「{}」で囲んで記述します。

## 例 1

SHA ハッシュ化パスワードの場合：

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 sn: Michaels
4 userpassword: {SHA}xcbdh46ngh37jsd0naSFDeDjAS30dm
5 objectclass: inetOrgPerson
```

## 例 2

SSHA ハッシュ化パスワードの場合：

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 sn: Michaels
4 userpassword: {SSHA}sGs948DFGkakdfkasDF34DF4dS3sk15DFS
5 objectclass: inetOrgPerson
```

## 例 3

Digest MD5 ハッシュ化パスワードの場合：

```
1 version: 1
2 dn: cn=Peter Michaels, ou=Artists, l=San Francisco, c=US
3 sn: Michaels
4 userpassword: {MD5}a451kSDF234SDFG62dsfsf2DG2QEvgdmnk430
5 objectclass: inetOrgPerson
```

## LDIF ファイルのデバッグ

LDIF ファイルで問題が発生した場合、次のことを考慮してください。

- ◆ [506 ページの「前方参照を有効化する」](#)
- ◆ [509 ページの「LDIF ファイルの構文のチェック」](#)
- ◆ [510 ページの「LDIF エラーファイルを使用する」](#)
- ◆ [511 ページの「LDAP SDK デバッグフラグを使用する」](#)

### 前方参照を有効化する

LDIF ファイルで、あるエントリを追加するレコードを、そのエントリのペアレントを追加するレコードの前に記述してしまう場合があります。この場合、LDAP サーバが新しいエントリを追加しようとする、そのエントリのペアレントが存在しないためエラーが発生します。

この問題は、前方参照の使用を有効にするだけで解決できます。前方参照の作成を有効にすると、エントリを作成するときそのペアレントがまだ存在していない場合でも、このペアレント用に前方参照というプレースホルダが作成されるため、エントリを正常

に作成できます。後の処理でペアレントが作成されると、前方参照は通常のエントリに変更されます。

LDIF のインポートが完了した後でも、1 つ以上の前方参照が残っている場合があります (たとえば、LDIF ファイルでエントリのペアレントが作成されなかった場合など)。この場合、前方参照は ConsoleOne および iManager に不明オブジェクトとして表示されます。前方参照エントリを検索することはできますが、前方参照エントリには属性も属性値もないため、objectClass 以外の属性を読み込むことはできません。ただし、前方参照の下に位置する実オブジェクトエントリ上では、すべての LDAP 操作が正常に機能します。

### 前方参照エントリを識別する


前方参照エントリは「不明」のオブジェクトクラスを持ち、また、内部 NDS EF\_REFERENCE エントリにフラグが設定されています。ConsoleOne および iManager では、「不明」のオブジェクトクラスを持つエントリは、中央に疑問符が表示される丸い黄色のアイコンで示されます。LDAP を使用して不明オブジェクトクラスのオブジェクトを検索することもできますが、現時点では LDAP からエントリフラグの設定にアクセスしてそれが前方参照エントリであることを確認する方法はありません。

### 前方参照エントリを通常オブジェクトへ変更する

(LDIF ファイルまたは LDAP クライアントリクエストなどを使用して) オブジェクトを作成するだけで、前方参照エントリを通常のオブジェクトに変更できます。eDirectory で作成するように指定したエントリが前方参照としてすでに存在する場合、eDirectory では既存の前方参照エントリが、作成を指定したオブジェクトに変換されます。

### Novell eDirectory インポート / エクスポート変換ウィザードを使用する

LDIF のインポート時に前方参照を有効にするには、次の手順に従ってください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルからデータをインポート] をクリックし、[次へ] をクリックします。
- 4 インポートするファイルのタイプに LDIF を指定します。
- 5 インポートするデータを含むファイルの名前と適切なオプションを指定して、[次へ] をクリックします。
- 6 データをインポートする LDAP サーバを指定します。
- 7 次の表に示すように、適切なオプションを追加します。


オプション	説明
サーバの DNS 名 / IP アドレス	相手 LDAP サーバの DNS 名または IP アドレス
ポート	相手 LDAP サーバのポート番号 (整数)
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法)

オプション	説明
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

8 [詳細設定] で、[前方参照を許可する] をクリックします。

9 [次へ] > [完了] の順にクリックします。

データをデータサーバへ移行するときに前方参照を有効にするには、次の手順に従ってください。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート 変換ウィザード] の順にクリックします。
- 3 [サーバ間でデータを移行] > [次へ] の順にクリックします。
- 4 移行するエントリが格納されている LDAP サーバを指定します。
- 5 次の表に示すように、適切なオプションを追加します。

オプション	説明
サーバの DNS 名 / IP アドレス	ソース LDAP サーバの DNS 名または IP アドレス
ポート	ソース LDAP サーバのポート番号 ( 整数 )
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法 )
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

6 [詳細設定] で、[前方参照を許可する] をクリックします。

7 [次へ] をクリックします。

8 移行するエントリの検索条件を次のように指定します。

オプション	説明
ベース DN	検索リクエストのベース識別名  このフィールドを指定しなかった場合、デフォルトのベース DN である ""( 空の文字列 ) が使用されます。
スコープ	検索リクエストのスコープ



オプション	説明
フィルタ	RFC 2254 準拠の検索フィルタ デフォルトは「objectclass=*」です。
属性	検索エン트리ごとに取得する属性

- 9 [次へ] をクリックします。
- 10 データを移行する LDAP サーバを指定します。
- 11 [次へ] > [完了] の順にクリックします。  
注: スキーマが各 LDAP サービスで整合性を保っていることを確認します。

### Novell インポート / エクスポート変換ユーティリティのコマンドラインインタフェースの使用

コマンドラインインタフェースで前方参照を有効にするには、-F LDAP ターゲットハンドラオプションを使用します。


詳細については、138 ページの「LDIF ターゲットハンドラのオプション」を参照してください。

### LDIF ファイルの構文のチェック

ファイル内のレコードを処理する前に、[操作を表示するが実行しない] LDIF ソースハンドラオプションを使用して LDIF ファイルの構文をチェックできます。

LDIF ソースハンドラでは、LDIF ファイル内のレコードを処理するとき常に構文がチェックされます。このオプションを使用すると、レコードの処理を無効にして構文を検証できます。

### Novell eDirectory インポート / エクスポート変換ウィザードを使用する

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory の保守] > [インポート / エクスポート変換ウィザード] の順にクリックします。
- 3 [ディスク上のファイルからデータをインポート] をクリックし、[次へ] をクリックします。
- 4 インポートするファイルのタイプに LDIF を指定します。
- 5 インポートするデータが含まれているファイルの名前を指定し、適切なオプションを指定します。
- 6 詳細設定で、[操作を実行せずに表示] をクリックし、[次へ] をクリックします。
- 7 データをインポートする LDAP サーバを指定します。
- 8 次の表に示すように、適切なオプションを追加します。

オプション	説明
サーバの DNS 名 / IP アドレス	相手 LDAP サーバの DNS 名または IP アドレス
ポート	相手 LDAP サーバのポート番号 (整数)

オプション	説明
DER ファイル	SSL 認証に使用するサーバキーが格納されている DER ファイルの名前
ログイン方法	「認証ログイン」または「匿名ログイン」( [ユーザ DN] フィールドに指定したエントリのログイン方法 )
ユーザ DN	サーバによって指定されたバインド操作に使用するエントリの識別名
パスワード	[ユーザ DN] フィールドで指定したエントリのパスワード属性

9 [次へ] > [完了] の順にクリックします。

### Novell インポート / エクスポート変換ユーティリティのコマンドラインインタフェースの使用

コマンドラインインタフェースで LDIF ファイルの構文をチェックするには、-n LDIF ソースハンドラオプションを使用します。

詳細については、[138 ページの「LDIF ソースハンドラのオプション」](#)を参照してください。

### LDIF エラーファイルを使用する

Novell インポート / エクスポート変換ユーティリティは、ターゲットハンドラによる処理に失敗したレコードをすべてリストした LDIF ファイルを自動的に作成します。ユーティリティによって生成された LDIF エラーファイルを編集してエラーを修正し、サーバに再適用することで、失敗したレコードに含まれているインポートまたはデータの移行を完了できます。

### Novell eDirectory インポート / エクスポートウィザードを使用する

この機能は ConsoleOne のみで使用できます。

- 1 ConsoleOne で、[ウィザード] > [NDS インポート / エクスポート] の順にクリックします。
- 2 実行するタスクをクリックします。
- 3 [詳細] をクリックします。
- 4 [ログファイル] フィールドに、出力メッセージ ( エラーメッセージを含む ) を記録するファイル名を指定します。
- 5 [失敗したレコードの LDIF 出力ファイル] フィールドに、失敗したエントリを LDIF 形式で出力するファイル名を指定します。

このファイルは、エラーの確認や訂正に使用できます。このファイルを修正 ( 訂正 ) して、もう一度ディレクトリに適用することもできます。

- 6 [閉じる] をクリックします。
- 7 表示される指示に従って、選択したタスクを完了します。

## Novell インポート / エクスポート変換ユーティリティのコマンドラインインタフェースの使用

コマンドラインユーティリティでエラーログオプションを設定するには、-l 一般オプションを使用します。

詳細については、[136 ページの「一般オプション」](#)を参照してください。

### LDAP SDK デバッグングフラグを使用する

一部の LDIF の問題を理解するには、LDAP クライアント SDK がどのように機能するかを理解する必要があります。LDAP ソースハンドラ、LDAP ターゲットハンドラ、またはその両方に、次のデバッグングフラグを設定できます。

値	説明
0x0001	LDAP ファンクションコールをトレースします。
0x0002	パケットに関する情報を出力します。
0x0004	引数に関する情報を出力します。
0x0008	接続情報を出力します。
0x0010	BER のエンコーディングおよびデコーディング情報を出力します。
0x0020	検索フィルタ情報を出力します。
0x0040	設定情報を出力します。
0x0080	ACL 情報を出力します。
0x0100	統計情報を出力します。
0x0200	追加の統計情報を出力します。
0x0400	シェル情報を出力します。
0x0800	解析情報を出力します。
0xFFFF (10 進数では、-1)	すべてのデバッグングオプションを有効にします。

この機能を有効にするには、LDAP ソースハンドラおよびターゲットハンドラで -e オプションを使用します。-e オプションに指定する整数の値は、LDAP SDK でさまざまな種類のデバッグ情報を有効にするビットマスクです。

詳細については、[139 ページの「LDAP ソースハンドラのオプション」](#) および [141 ページの「LDAP ターゲットハンドラのオプション」](#)を参照してください。

## LDIF を使用してスキーマを拡張する

LDIF では LDAP 更新操作を表すことができるので、LDIF を使用してスキーマを変更できます。

### 新しいオブジェクトクラスを追加する

クラスを追加するには、単に、NDSObjectClassDescription の仕様に従った属性値を subschemaSubentry の objectClasses 属性に追加します。

```
NDSObjectClassDescription = "(" whsp
    numericoid whsp
    [ "NAME" qdescrs ]
    [ "DESC" qdstring ]
    [ "OBSOLETE" whsp ]
    [ "SUP" oids ]
    [ ( "ABSTRACT" / "STRUCTURAL" / "AUXILIARY" ) whsp ]
    [ "MUST" oids ]
    [ "MAY" oids ]
    [ "X-NDS_NOT_CONTAINER" qdstrings ]
    [ "X-NDS_NONREMOVABLE" qdstrings ]
    [ "X-NDS_CONTAINMENT" qdstrings ]
    [ "X-NDS_NAMING" qdstrings ]
    [ "X-NDS_NAME" qdstrings ]
whsp ")"
```

次の LDIF ファイルの例では、person objectClass をスキーマに追加します。

```
1 version: 1
2 dn: cn=schema
3 changetype: add
4 objectClasses: ( 2.5.6.6 NAME 'person' DESC 'Standard
5   ObjectClass' SUP ndsLoginProperties STRUCTURAL MUST
6   (cn $ sn) MAY (description $ seeAlso $ telephoneNum
7   ber $ fullName $ givenName $ initials $ uid $ userPa
8   ssword) X-NDS_NAMING ('cn' 'uid') X-NDS_CONTAINMENT
9   ('organization' 'organizationalUnit' 'domain') X-NDS
10  _NAME 'Person' X-NDS_NOT_CONTAINER '1' X-NDS_NONREMO
11  VABLE '1')
12
```

#### 必須属性

必須属性は、オブジェクトクラス記述の MUST セクションにリストします。person オブジェクトクラスの場合、必須属性は cn と sn です。

#### オプション属性

オプション属性のリストは、オブジェクトクラス記述の MAY セクションに記述します。person オブジェクトクラスのオプション属性は、description、seeAlso、telephoneNumber、fullName、givenName、initials、uid、および userPassword です。

**注:** userPassword 属性は、オプション (MAY) 属性には使用できません。この LDIF 形式を使用して、新しい objectClass でこの属性を必須 (MUST) 属性に使用してスキーマを拡張しようとしても、操作は失敗します。

#### 包含ルール

定義されているオブジェクトクラスを包含するオブジェクトクラスは、オブジェクトクラス記述の X-NDS\_CONTAINMENT セクションで指定します。person オブジェクトクラスを包含するオブジェクトクラスは、organization、organizationalUnit、および domain です。

## 新しい属性を追加する

属性を追加するには、NDSAttributeTypeDescriptionの仕様に従って属性値をsubschemaSubentryの attributes 属性に追加します。

```
NDSAttributeTypeDescription = "(" whsp
    numericoid whsp ; AttributeType identifier
    [ "NAME" qdescrs ] ; name used in AttributeType
    [ "DESC" qdstring ] ; description
    [ "OBSOLETE" whsp ]
    [ "SUP" woid ] ; derived from this other AttributeType
    [ "EQUALITY" woid ] ; Matching Rule name
    [ "ORDERING" woid ] ; Matching Rule name
    [ "SUBSTR" woid ] ; Matching Rule name
    [ "SYNTAX" whsp noidlen whsp ] ; Syntax OID
    [ "SINGLE-VALUE" whsp ] ; default multi-valued
    [ "COLLECTIVE" whsp ] ; default not collective
    [ "NO-USER-MODIFICATION" whsp ] ; default user modifiable
    [ "USAGE" whsp AttributeUsage ] ; default userApplications
    [ "X-NDS_PUBLIC_READ" qdstrings ]
        ; default not public read ('0')
    [ "X-NDS_SERVER_READ" qdstrings ]
        ; default not server read ('0')
    [ "X-NDS_NEVER_SYNC" qdstrings ]
        ; default not never sync ('0')
    [ "X-NDS_NOT_SCHED_SYNC_IMMEDIATE" qdstrings ]
        ; default sched sync immediate ('0')
    [ "X-NDS_SCHED_SYNC_NEVER" qdstrings ]
        ; default schedule sync ('0')
    [ "X-NDS_LOWER_BOUND" qdstrings ]
        ; default no lower bound('0')
        ;(upper is specified in SYNTAX)
    [ "X-NDS_NAME_VALUE_ACCESS" qdstrings ]
        ; default not name value access ('0')
    [ "X-NDS_NAME" qdstrings ] ; legacy NDS name
whsp ")"
```

次の LDIF ファイルの例では、title 属性タイプをスキーマに追加します。

```
1 version: 1
2 dn: cn=schema
3 changetype: add
4 attributeTypes: ( 2.5.4.12 NAME 'title' DESC 'Standa
5 rd Attribute' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{
6 64} X-NDS_NAME 'Title' X-NDS_NOT_SCHED_SYNC_IMMEDIA
7 TE '1' X-NDS_LOWER_BOUND '1')
8
```

### 単一値と複数値

属性は、明示的に単一値として定義されない限り、デフォルトでは複数値です。次の LDIF ファイルの例では、SYNTAX セクションの後に SINGLE-VALUE キーワードを追加することによって、title を単一値として定義しています。

```
1 version: 1
2 dn: cn=schema
3 changetype: add
4 attributeTypes: ( 2.5.4.12 NAME 'title' DESC 'Standa
5 rd Attribute' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15{
6 64} SINGLE-VALUE X-NDS_NAME 'Title' X-NDS_NOT_SCHED
7 _SYNC_IMMEDIATE '1' X-NDS_LOWER_BOUND '1')
8
```

## 既存のオブジェクトクラスへオブション属性を追加する

新しいスキーマエレメントを追加する場合は問題ありませんが、通常、既存のスキーマエレメントを変更または拡張する場合には注意が必要です。すべてのスキーマエレメントはOIDによって固有に識別されるため、標準スキーマエレメントを拡張すると、元のOIDを使用する場合でも実際にはそのエレメントに対して2つめの定義が作成されます。このため、不整合が発生することがあります。

スキーマエレメントの変更が必要な場合もあります。たとえば、開発しながらスキーマエレメントを洗練していくときに、新しいスキーマエレメントの拡張または修正が必要な場合があります。次のような場合は、クラスに直接新しい属性を追加せずに、通常は補助クラスのみを使用します。

- ◆ 既存のオブジェクトクラスに新しい属性を追加する場合
- ◆ 既存のオブジェクトクラスのサブクラスを作成する場合

## 補助クラスを追加または削除する

次のサンプル LDIF ファイルは、2 つの新しい属性、およびこの新しい属性に付随する補助クラスを作成してから、inetOrgPerson エントリをエントリのオブジェクトクラスとして補助クラスと補助クラスの属性値に追加します。

```
version: 1
# Add an attribute to track a bear's hair.The attribute is
# multi-valued, uses a case ignore string syntax,
# and has public read rights
# Values may include:long hair, short, curly, straight,
# none, black, and brown
# X-NDS_PUBLIC_READ '1' The 1 allows public read,
# 0 denies public read
dn: cn=schema
changetype: modify
add: attributeTypes
attributeTypes: ( 2.16.840.1.113719.1.186.4.10 NAME
'bearHair' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
X-NDS_PUBLIC_READ '1' )

# add an attribute to store a bear's picture
dn: cn=schema
changetype: modify
add: attributeTypes
attributeTypes: ( 2.16.840.1.113719.1.186.4.11 NAME
'bearPicture' SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE )

# create an Auxiliary class for the bearfeatures
dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: (2.16.840.1.113719.1.186.6.101 NAME
'bearFeatures' MAY (bearHair $ bearPicture) AUXILIARY)

# now create a user named bobby
dn: cn=bobby,o=bearcave
changetype: add
cn: bobby
sn: bear
givenName: bobby
bearHair: Short
```

```

bearHair: Brown
bearHair: Curly
bearPicture: <file:///c:/tmp/alien.jpg>
objectClass: top
objectClass: person
objectClass: inetOrgPerson
objectClass: bearFeatures

# now create a person named john that will later be changed
# into a bear when bearFeatures is added to its objectClass
# list
dn: cn=john,o=bearcave
changetype: add
cn: John
sn: bear
givenName: john
objectClass: top
objectClass: person
objectClass: inetOrgPerson

# now morph john into a bear by adding bearFeatures
dn: cn=john,o=bearcave
changetype: modify
add: objectClass
objectClass: bearFeatures
-
add: bearHair
bearHair: long
bearHair: black
#bearPicture: <file:///c:/tmp/john.jpg>
-

# to morph john back to a person, simply delete the
# objectClass bearFeatures
dn: cn=john,o=bearcave
changetype: modify
delete: objectClass
objectClass: bearFeatures

```

補助クラスの削除にあたって、**objectClass** リストから補助クラスを削除する場合には、補助クラスに関連するすべての値を削除する必要はありません。この処理は **eDirectory** によって自動的に行われます。

補助クラスに **MUST** 属性がある場合、補助クラスを **objectClass** リストへ追加する変更操作でもこれらの属性を指定する必要があります。これらの属性が指定されていない場合、変更は失敗します。

### XML 解析で発生する既知の問題

個々のレコードが XML ファイルで指定されたすべての XML ルールを遵守していない場合、LDIF レコード (LDAP サーバで生成された LDIF 形式またはレコード) の XML 処理は成功しません。

# Linux、Solaris、AIX、および HP-UX での eDirectory のトラブルシューティング

このセクションには、Linux、Solaris、AIX、および HP-UX プラットフォーム上の eDirectory のトラブルシューティングについての情報が含まれます。

- ◆ [516 ページの「eDirectory への繰り返しログイン」](#)
- ◆ [516 ページの「Novell パブリックキーインフラストラクチャサービス」](#)
- ◆ [517 ページの「UNIX 上の NMAS」](#)
- ◆ [517 ページの「LDAP サービス」](#)
- ◆ [518 ページの「Novell インポート / エクスポート変換ユーティリティ」](#)
- ◆ [519 ページの「ndsmerge ユーティリティ」](#)
- ◆ [519 ページの「ndstrace ユーティリティ」](#)
- ◆ [519 ページの「ndsbackup ユーティリティ」](#)
- ◆ [519 ページの「インストールと設定」](#)
- ◆ [521 ページの「ndsrepair を使用する」](#)
- ◆ [528 ページの「ndstrace の使用」](#)

## eDirectory への繰り返しログイン

eDirectory に繰り返しログインする場合、すべての使用可能なメモリを使用できます。ndsmonitor を使用してログイン更新属性を無効にすると、この問題を解決できます。

## Novell パブリックキーインフラストラクチャサービス

### PKI 操作が機能しない

ConsoleOne または iManager で PKI 操作が機能しない場合、Novell PKI サービスが Linux、Solaris、AIX、または HP-UX ホストで実行されていないことが考えられます。「[npki - 1](#)」と入力して PKI サービスを開始してください。

証明書を作成できない場合は、NICI モジュールが正しくインストールされているか確認する必要があります。[82 ページの「サーバ上の NICI モジュールを初期化する」](#)を参照してください。NICI が初期化されているかを確認する方法については、[82 ページの「サーバ上に NICI がインストールおよび初期化されているかどうかを確認する」](#)を参照してください。

### Netscape アドレスブックの LDAP 検索の失敗


エクスポートバージョンの Netscape ブラウザおよび LDAP サーバオブジェクトに関連付けられた 512 ビットよりも大きな KMO キーサイズを使用している場合、Netscape アドレスブックの LDAP 検索が失敗することがあります。

この場合は、米国版の Netscape ブラウザを使用してください。



## 重要なレプリカというエラーコードが表示され、既存の eDirectory オブジェクトの別のサーバへの移動が失敗した後に、マルチサーバツリー内でツリーキーサーバとして機能している eDirectory サーバの削除

この操作を完了するには、[セキュリティコンテナ] > [KAP] 以下にある W0 オブジェクトで、キーサーバ DN 属性をこのサーバからツリーキーをダウンロードしたツリー内にある別のサーバに変更します。

- 1 Novell iManager で、[役割およびタスク] ボタン  をクリックします。
- 2 [eDirectory 管理] > [オブジェクトの変更] の順にクリックします。
- 3 W0 オブジェクト名およびコンテキスト (通常は、W0.KAP.Security) を指定して、[OK] をクリックします。
- 4 [値がある属性] カラムで、[NDSPKI:SD キーサーバ DN] を選択してから、[編集] をクリックします。
- 5 [セキュリティドメインキーサーバの DN] フィールドで別のサーバの名前とコンテキストを指定してから、[OK] をクリックします。
- 6 [適用] をクリックし、[OK] をクリックします。

## CA を保持している eDirectory サーバのアンインストール中に、サーバに作成された KMO がツリー内の別のサーバに移動されて無効になる

この場合、ツリーの CA および KMO をもう一度作成する必要があります。詳細については、[83 ページの「組織の認証局オブジェクトを作成する」](#) および [84 ページの「サーバ認証オブジェクトを作成する」](#) を参照してください。

ツリーの CA を保持する eDirectory は、アンインストールしないことをお勧めします。

## UNIX 上の NMAS

### どのメソッドを使用してもログインできない

NMAS をインストールおよび設定した後で、NDS サーバを再起動します。

メソッドの以前のインスタンスをアンインストールしてからメソッドを再インストールした後で、NDS サーバを再起動します。

### ICE ユーティリティを使用して追加したユーザが、簡易パスワードを使用してログインできない

Novell インポート / エクスポート変換ユーティリティを使用して簡易パスワードを使用するユーザを追加する場合は、-l オプションを使用します。

## LDAP サービス

このセクションでは、LDAP Services for eDirectory で起こりうる一般的な問題と、その解決法について説明します。

LDAP クライアントからリクエストが発行される前に、LDAP サーバが起動していることを確認します。/var/nds/nds.log ファイル内に次のメッセージがある場合は、LDAP サーバが起動していることが確認できます。

```
LDAP v3 for Novell eDirectory 8.7.3 started
```

詳細については、[12 章 299 ページの「LDAP Services for Novell eDirectory の環境設定」](#) を参照してください。

## LDAP クライアントが LDAP Services for eDirectory にバインドできない

LDAP クライアントを LDAP Services for eDirectory にバインドできない場合は、次の内容を確認します。

- ユーザは正しいユーザ名とパスワードを入力しているか。
- ユーザは名前を LDAP 形式で入力しているか。
- パスワードの有効期限が切れていないか。
- サーバが再設定されたか。

## LDAP サーバで新しい設定が使用されていない

LDAP サーバ設定の更新処理は、現在バインドされている LDAP クライアントの影響を受けることがあります。

設定の変更は動的に更新されます。LDAP サーバでは、設定の変更が定期的 (30 分ごと) に確認されます。変更が検出されると、再設定処理が実行されている間、新しいクライアントは LDAP サーバにバインドできません。

LDAP サーバでは、現在バインドしているクライアントの新しい LDAP リクエストの処理が中止され、実行中の LDAP リクエストが完了するまで待機してから、設定が更新されます。

ndsmerge ユーティリティを使用してツリー名を変更した場合、LDAP 操作は失敗します。適切に操作が完了するには、ツリー名の変更後に LDAP サーバをリフレッシュまたは再起動する必要があります。

## 安全な LDAP 接続を確立できない

次を確認します。

- LDAP サーバの認証局および暗号化キーオブジェクト (KMO) が作成されている。
- KMO が LDAP サーバに関連付けられている。
- 指定された認証局署名有効期限が切れていない。システムの日付が有効期限日を過ぎているかどうかを確認します。
- 安全な LDAP ポートで LDAP サーバがリスンしている。デフォルトは 636 です。
- iManager で LDAP サーバオブジェクトの SSL が有効になっている。

詳細については、[82 ページの「Linux、Solaris、AIX、および HP-UX システムでの eDirectory 操作に関するセキュリティを確保する」](#)を参照してください。

## Novell インポート / エクスポート変換ユーティリティ

LDAP サーバがリフレッシュまたはアンロードされると、Novell インポート / エクスポート変換操作の実行中に「LBURP operation is timed out」というメッセージが Novell インポート / エクスポート変換画面上に表示されます。LBURP 操作がタイムアウトした場合、サーバは後で復元されます。

## ndsmerge ユーティリティ

マージ操作後は、PKI サーバはアクティブになっていません。したがって、`upki -l` コマンドを使用して再起動する必要があります。

異なるバージョンの製品では、マージ操作が成功しないことがあります。サーバで古いバージョンの NDS または eDirectory が実行されている場合は、最新バージョンの eDirectory にアップデートしてからマージ操作を続行してください。

ツリーに付随するよく似た名前前のコンテナがソースツリーおよびターゲットツリーの両方にある場合、2 つのツリーのマージは成功しません。どちらかのコンテナ名を変更してから、マージ操作を続行してください。

結合操作の実行中に、「-611 不正な包含です」というエラーメッセージが表示される場合があります。ndsrepair(1) を実行してスキーマを変更します。次に、ndsrepair -s を実行し、[オプションスキーマ拡張機能] を選択します。

## ndstrace ユーティリティ

ndstrace(1) 画面をオンにすると、参照リンクに対するプライマリオブジェクトが不正であることを示すエラーメッセージが表示される場合があります。eDirectory が正常に機能している場合は、このメッセージを無視してください。

## ndsbackup ユーティリティ

eDirectory のバックアップの実行中に、「NDS エラー: NDS サーバへの接続に失敗しました」というエラーメッセージが表示される場合があります。これは、デフォルトのポート 524 以外のポートでリッスンしている eDirectory が原因である可能性があります。コマンドラインで、eDirectory が設定されているポート番号を入力します。たとえば、eDirectory がポート番号 1524 に設定されている場合は次のように入力します。

```
ndsbackup sR 164.99.148.82:1524
```

## インストールと設定

### インストールの失敗

- ◆ /var/adm/messages ディレクトリで、次のエラーメッセージを確認します。  
SLP マルチキャストアドレスにバインドできませんマルチキャストルートが追加されていない可能性があります。  
Linux または Solaris のコンピュータがマルチキャストルートアドレスに設定されていない場合、このメッセージが表示されます。  
マルチキャストルートアドレスを追加して、slpuasa デーモンを再起動します。
- ◆ インストール中に「-632: エラーの説明 システムエラーです」というエラーメッセージが表示されたら、インストール処理を終了します。  
/etc/nds.conf ファイルで n4u.base.slp.max-wait パラメータを 50 などの大きい値に設定してから、インストール処理をもう一度開始します。
- ◆ eDirectory を NetWare 5.1 ツリーにインストールする場合、eDirectory Master を NetWare 5.1 Support Pack 5 以降にアップグレードします。  
詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「Novell eDirectory の NetWare へのインストールまたはアップグレード」を参照してください。

- ◆ eDirectory for Solaris 2.0のインストールプログラムをアップグレードできなかった場合、次のインストールが中断される可能性があります。

/var/nds/.n4s\_upgrade ファイルを削除して、もう一度インストールしてください。

- ◆ インストール中に「ツリー名が見つかりません」というエラーメッセージが表示された場合は、次の手順を実行します。

**1** 製品をインストールしようとしている Solaris ホストでマルチキャストルーティングが有効に設定されていることを確認します。

**2** ツリーパーティションのマスタサーバの IP アドレスを指定します。

## インストールに長い時間がかかる

eDirectory を既存のツリーにインストールする場合に、インストールの完了までに長時間かかるときは、サーバの dstrace 画面を確認してください。「-625 転送できません」というメッセージが表示された場合は、アドレスキャッシュをリセットする必要があります。

アドレスキャッシュをリセットするには、システムコンソールで次のコマンドを入力します。

```
set dstrace = *A
```

## WAN 上の既存のツリーにインストールできない

eDirectory を WAN 上の Linux または Solaris システムにインストールするには、NetWare 5 以降のサーバが必要です。

- 1** サーバコンソールで次のコマンドを入力し、NetWare サーバ上でディレクトリエージェント (DA) を実行します。

```
slpda
```

- 2** マスタレプリカを保持するサーバで、slpuasa.conf ファイルの DA\_ADDR パラメータを次のように編集します。

```
DA_ADDR = IP_address_of_the_NetWare_server_where_the_DA_is_running
```

- 3** slpuasa デーモンを再起動します。

- 4** WAN 上の Linux または Solaris システムに eDirectory をインストールします。

**4a** nds-install を実行して、製品パッケージを追加します。

製品の設定は行わないでください。詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「Linux、Solaris、AIX、および HP-UX 用 Novell eDirectory パッケージ」を参照してください。

**4b** /etc/nds.conf ファイルを編集し、次のパラメータを追加します。

```
n4u.uam.ncp-retries = 5
n4u.base.slp.max-wait = 20
```

**4c** /etc/slpuasa.conf ファイルを編集し、次のパラメータを追加します。

```
DA_ADDR = IP_address_of_the_NetWare_server_where_the_DA_is_running
```

**4d** ndsconfig を実行して eDirectory を設定します。

## ndsrepair を使用する

このセクションでは次について説明します。

- ◆ [521 ページの「構文」](#)
- ◆ [528 ページの「ndsrepair のトラブルシューティング」](#)

サーバコンソールでの ndsrepair ユーティリティの用途は次のとおりです。

- ◆ 不正なレコード、スキーマの不一致、不正なサーバアドレス、外部参照など、eDirectory の問題を修正する。
- ◆ eDirectory スキーマに詳細な変更を加える。
- ◆ eDirectory データベースで次の操作を実行する。
  - ◆ データベースの終了やデータベースの介入を伴わない、自動的なデータベースの構造チェック。
  - ◆ データベースインデックスを確認する。
  - ◆ データベースの終了やユーザのロックアウトを伴わず、データベースを修復する。
  - ◆ 空のレコードを破棄して、空き領域を増量する。

### 構文

ndsrepair を実行するには、次の構文を使用します。

```
ndsrepair {-U| -P| -S| -C| -E| -N| -T| -J entry_id}
[-A yes|no] [-O yes|no] [-F filename] [-Ad]
```

または

```
ndsrepair -R [-l yes|no [-u yes|no] [-m yes|no] [-i yes|no] [-f yes|no] [-d
yes|no] [-t yes|no] [-o yes|no] [-r yes|no] [-v yes|no] [-c yes|no] [-A
yes|no] [-O yes|no] [-F filename]
```

**重要:** -Ad オプションは、Novell サポート担当者からの事前の指示がない限り使用しないでください。

### ndsrepair のオプション

オプション	説明
-U	[標準修復] オプションです。ユーザの操作なしに ndsrepair を実行または終了します。Novell サポート担当者から特定の操作を手動で実行するように指示された場合を除き、この修復方法の使用をお勧めします。修復が完了したらログファイルをチェックして、ndsrepair で変更された内容を確認します。
-P	[レプリカ操作とパーティション操作] オプションです。現在のサーバの eDirectory データベースファイルにレプリカが保存されているパーティションが表示されます。[レプリカオプション] メニューには、「レプリカの修復」、「パーティション操作のキャンセル」、「同期のスケジューリング」、および「ローカルレプリカをマスターレプリカとして指定」を実行するオプションがあります。  詳細については、 <a href="#">524 ページの「[レプリカ操作とパーティション操作] オプション」</a> を参照してください。

オプション	説明
-S	[グローバルスキーマの操作] オプションです。このオプションには、このサーバのスキーマをツリーオブジェクトのマスタに準拠させるのに必要なスキーマ操作がいくつか含まれています。ただし、これらの操作は必要なときだけ使用してください。スキーマは、ローカル修復操作および標準修復操作によってすでに検査されています。
-C	[外部参照オブジェクトのチェック] オプションです。各外部参照オブジェクトをチェックして、そのオブジェクトを含むレプリカがあるかどうかを調べます。オブジェクトのあるパーティションレプリカを含むすべてのサーバがアクセス不能の場合、オブジェクトは見つかりません。オブジェクトが見つからない場合、警告が表示されます。
-E	[レプリカ同期のレポート] オプションです。現在のサーバ上にレプリカを持つすべてのパーティションのレプリカ同期ステータスをレポートします。この操作により、パーティションのレプリカを保持する各サーバ上のレプリカの Tree オブジェクトから同期ステータス属性が読み込まれます。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。12 時間以内に同期が完了していない場合は、警告メッセージが表示されます。
-N	[このデータベースに認識されているサーバ] オプションです。ローカル eDirectory データベースに認識されているすべてのサーバが表示されます。現在のサーバにツリーパーティションのレプリカがある場合、このサーバには eDirectory ツリー内のすべてのサーバのリストが表示されます。サーバオプションを実行するサーバを 1 つ選択します。
-J	ローカルサーバ上の 1 つのオブジェクトを修復します。修復するオブジェクトのエントリ ID(16 進形式) を指定する必要があります。破損している 1 つの特定のオブジェクトを修復するには、[標準修復 (-U)] オプションの代わりに、このオプションを使用できます。データベースのサイズによっては、[標準修復] オプションの完了に何時間もかかる場合があります。このオプションを使用して、時間を節約することができます。
-T	[時刻同期] オプションです。ローカル eDirectory データベースに登録されているすべてのサーバにアクセスして、各サーバの時刻同期ステータスの情報を要求します。このサーバに Tree パーティションのレプリカがある場合は、eDirectory ツリー内のすべてのサーバがポーリングされます。各サーバ上で実行されている eDirectory のバージョンもレポートされます。
-A	既存のログファイルに付加します。既存のログファイルに情報が追加されます。このオプションは、デフォルトで有効に設定されています。
-O	出力をファイルに記録します。このオプションは、デフォルトで有効に設定されています。
-F filename	出力を指定したファイルに記録します。
-R	[ローカルデータベースの修復] オプションです。ローカル eDirectory データベースを修復します。eDirectory でオープンおよびアクセスできるように、修復操作を使用してローカルデータベースの矛盾を解決します。このオプションには、データベースの修復操作を容易にするサブオプションがあります。このオプションにはファンクション修飾子があります。ファンクション修飾子については、 <a href="#">523 ページの「-R オプションで使用するファンクション修飾子」</a> で説明されています。

## -R オプションで使用するファンクション修飾子

Modifier	説明
-l	修復操作中に eDirectory データベースをロックします。
-u	修復操作中に一時 eDirectory データベースを使用します。
-m	修復されていない元のデータベースを維持します。
-i	eDirectory データベース構造とインデックスをチェックします。
-f	データベースの空き領域を増やします。
-d	データベース全体を再構築します。
-t	ツリー構造のチェックを実行します。データベース内での接続状況が正しいかどうかを調べるため、ツリー構造のリンクをすべてチェックするには、「はい」を設定します。チェックを省略するには、「いいえ」を設定します。 デフォルト値は「はい」です。
-o	オペレーショナルスキーマを再構築します。
-r	すべてのローカルレプリカを修復します。
-v	ストリームファイルを確認します。
-c	ローカル参照をチェックします。

## グローバルスキーマ操作

ndsrepair -S( [-Ad] 詳細設定スイッチ ) オプションを使用して、実行できるすべてのスキーマ操作のリストを表示できます。次の表では、使用可能なオプションについて説明します。

オプション	説明
マスタサーバからスキーマを要求	このサーバのスキーマを同期するようツリーのルートのマスタレプリカに要求します。スキーマの変更内容は、ツリーオブジェクトのマスタレプリカからこのサーバに 24 時間以内に伝えられます。すべてのサーバがマスタレプリカのスキーマを要求すると、ネットワークトラフィックが増加します。
ローカルスキーマのリセット	ローカルスキーマのタイムスタンプをクリアし、インバウンドスキーマの同期を要求するスキーマリセット機能が呼び出されます。ツリーパーティションのマスタレプリカから実行する場合、このオプションは使用できません。これは、ツリー内のすべてのサーバが同時にリセットされないようにするためです。
Post NetWare 5 スキーマの更新	post-NetWare 5 DS の変更に伴い、互換性を確保するためにスキーマを拡張および変更します。このオプションを使用するには、ndsrepair が実行されているサーバがツリーパーティションのレプリカを保持していて、レプリカのステータスがオンである必要があります。

オプション	説明
オプションスキーマ拡張機能	<p>包含やその他の拡張機能をスキーマに追加し、変更します。このオプションを使用するには、このサーバがツリーパーティションのレプリカを保持し、レプリカのステータスがオンである必要があります。さらに、ツリー内のすべての NetWare 4 サーバに次のバージョンの eDirectory がインストールされている必要があります。</p> <ul style="list-style-type: none"> <li>◆ NetWare 4.10 サーバでは、NDS バージョン 5.17 以降が必要です。</li> <li>◆ NetWare 4.11/4.2サーバでは、NDSバージョン6.03以降が必要です。</li> </ul> <p>旧バージョンの NDS では、これらの変更内容を同期できません。</p>
リモートスキーマのインポート (詳細設定スイッチオプション)	<p>現在のツリーのスキーマに追加するスキーマを含む eDirectory ツリーを選択します。ツリーを選択すると、ツリーパーティションのマスタレプリカを保持するサーバに接続されます。現在のツリー上にあるスキーマを拡張するには、そのサーバのスキーマが使用されます。</p>
新規エポックの宣言 (詳細設定スイッチオプション)	<p>新しいスキーマエポックを宣言すると、ツリーパーティションのマスタレプリカにアクセスし、そのサーバで宣言されるスキーマの無効なタイムスタンプが修復されます。他のすべてのサーバは、修復されたタイムスタンプを保持する新しいスキーマのコピーを受け取ります。受け取る側のサーバが新規エポック内に存在しなかったスキーマを含む場合は、古いスキーマを使用するオブジェクトおよび属性が「不明」オブジェクトクラスまたは属性に変更されます。</p>

## [レプリカ操作とパーティション操作] オプション

サーバに保存された各レプリカの情報を表示するには、次のコマンドを入力します。

```
ndsrepair -P
```

必要なレプリカを選択します。次のオプションが表示されます。

- ◆ すべてのレプリカの修復

レプリカテーブルに表示されたレプリカをすべて修復します。

- ◆ 選択したレプリカの修復

レプリカテーブルに表示されているレプリカのうち、選択したレプリカのみを修復します。

**重要:** レプリカの修復では、レプリカ内の各オブジェクトとスキーマとの整合性のチェック、データと属性構文との整合性のチェックが行われます。レプリカに関連する他の内部データ構造もチェックされます。過去 30 分間にローカル eDirectory データベースを修復していない場合は、レプリカを修復する前にローカル eDirectory データベースを修復する必要があります。

- ◆ 即時同期のスケジューリング

すべてのレプリカの同期を直ちに行うようにスケジューリングします。この操作は、同期が予定通り実行されるまで待たずに、その同期プロセスの eDirectory 情報を ndstrace 画面で参照したい場合に便利です。



- ◆ パーティション操作のキャンセル

選択したパーティション上でのパーティション操作をキャンセルします。このオプションは、サーバの検出不可や通信リンクの不良など、eDirectory ツリーに問題があるために操作が完了できない場合に必要になります。操作がある程度以上進行していると、キャンセルできない場合があります。

- ◆ このサーバを新しいマスタレプリカに設定

選択したパーティションのローカルレプリカをマスタレプリカとして設定します。元のマスタレプリカが失われた場合、このオプションを使用して新しいマスタレプリカを設定します。

- ◆ すべてのサーバの同期ステータスのレポート

現在のサーバ上にあるすべてのパーティションのレプリカ同期ステータスをレポートします。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。

- ◆ すべてのサーバ上のレプリカの同期

選択したパーティションのレプリカを保持するすべてのサーバ上で、完全な同期ステータスを確保します。このオプションを使用して、パーティションの状態を確認できます。該当するパーティションのレプリカを保持するすべてのサーバが正常に同期されていれば、そのパーティションは正常に機能していると判断できます。各サーバは、レプリカリング内の他のすべてのサーバに対して即時同期を実行します。サーバは、サーバ自体には同期されません。したがって、現在のサーバが所有するレプリカのステータスは「ホスト」と表示されます。

- ◆ すべてのレプリカのリングの修復

レプリカテーブルに表示されたすべてのレプリカのレプリカリングを修復します。

- ◆ 選択したレプリカのリングの修復

レプリカテーブルから選択したレプリカのレプリカリングを修復します。

**重要:** レプリカリングの修復では、指定したパーティションのレプリカを含む各サーバ上のレプリカリング情報がチェックされ、リモート ID 情報が検証されます。過去 30 分間にローカル eDirectory データベースを修復していない場合は、すべてのリングまたは選択したリングを修復する前に、ローカル eDirectory データベースを修復する必要があります。ローカルデータベースを修復するには、-R オプションを使用します。詳細については、[522 ページの「-R」](#)を参照してください。

- ◆ レプリカリングの表示

選択したパーティションのレプリカを保持するすべてのサーバのリストを表示します。このサーバセットをレプリカリングと呼びます。レプリカリングのリストには、レプリカのタイプに関する情報およびリング内にある各サーバの現在のステータスが表示されます。レプリカリングを表示してからサーバを選択すると、サーバオプションが表示されます。

### サーバオプション

- ◆ 選択したサーバの同期ステータスのレポート

選択したサーバ上にレプリカを保持する選択したパーティションのレプリカ同期ステータスをレポートします。この操作により、パーティションのレプリカを保持する各サーバ上のレプリカルートオブジェクトから、同期ステータス属性が読み込まれます。レポートには、すべてのサーバに対して正常に同期が行われた最終時刻と、最終同期以降発生したエラーが表示されます。12 時間以内に同期が完了していない場合は、警告メッセージが表示されます。

- ◆ 選択したサーバのレプリカの同期

選択したパーティションのレプリカを保持する選択したサーバ上で、完全な同期ステータスを確保します。このオプションを使用して、パーティションの状態を確認できます。該当するパーティションのレプリカを持つサーバが正常に同期されていれば、そのパーティションが正常に機能していると判断できます。該当するサーバは、直ちにレプリカリング内の他のすべてのサーバに同期されます。サーバは、それ自体には同期されません。したがって、現在のサーバが所有するレプリカのステータスは「ホスト」として表示されます。

- ◆ リング内のすべてのレプリカにすべてのオブジェクトを送信

レプリカリング内で選択したサーバから、選択したパーティションのレプリカを保持する他のすべてのサーバに、すべてのオブジェクトを送信します。この操作によって大量のネットワークトラフィックが生成されることがあります。レプリカリング内で選択したサーバ上の選択したパーティションのレプリカが、レプリカリング内の他のすべてのサーバと同期されていることを確かめるには、このオプションを使用します。該当するパーティションのサブオーディネートリファレンスレプリカのみを保持するサーバではこの操作は実行できません。

- ◆ マスタからこのレプリカへのすべてのオブジェクトの受信

選択したサーバ上のレプリカで、マスタレプリカのすべてのオブジェクトを受信します。この操作によって大量のネットワークトラフィックが生成されることがあります。レプリカリング内で選択したサーバ上の選択したパーティションのレプリカが、マスタレプリカと同期していることを確かめるには、このオプションを使用します。マスタレプリカのみを保持するサーバ上ではこの操作は実行できません。

- ◆ サーバのフルネームを表示

このオプションは、サーバ名が長すぎてサーバテーブル内にすべてを表示できない場合に、完全なサーバ名を表示するために使用します。

- ◆ レプリカリングからのサーバの削除

(詳細設定スイッチオプション) 現在のサーバに保存されている特定のレプリカから、選択したサーバを削除します。レプリカリング内に表示されるサーバがすでに eDirectory ツリーの一部分ではない場合や、パーティションのレプリカを保持していない場合は、iManager を使用してこのサーバオブジェクトを削除します。サーバオブジェクトが削除されると、そのオブジェクトは最終的にレプリカリングから除外されます。

**警告：** この操作を誤用すると、eDirectory データベースで致命的な破損が生じることがあります。Novell サポート担当者の指示がない限り、このオプションは使用しないでください。

- ◆ フルパーティション名の表示

このオプションは、パーティション名が長すぎてレプリカテーブル内にすべてを表示できない場合に、完全なパーティション識別名を確認するために使用します。

- ◆ タイムスタンプの修復と新規エポックの宣言

(詳細設定スイッチオプション) 選択したパーティションのレプリカをすべて最新版に更新するために、マスタレプリカの新しい参照ポイントを指定します。この操作は、常にパーティションのマスタレプリカ上で実行されます。マスタレプリカは、このサーバのローカルレプリカにある必要はありません。オブジェクトが作成または変更されるとタイムスタンプが設定されます。これらのタイムスタンプは固有である必要があります。マスタレプリカのすべてのタイムスタンプが検査されます。タイムスタンプが現在のネットワーク時間より遅れている場合は、新しいタイムスタンプに置き換えられます。

- ◆ 選択したレプリカの削除

( 詳細設定スイッチオプション ) このサーバ上の選択したレプリカを削除します。このオプションの使用はお勧めできません。このオプションは、他のユーティリティを使用してレプリカを削除できない場合にのみ使用してください。

- ◆ 不明リーフオブジェクトの削除

( 詳細設定スイッチオプション ) 不明オブジェクトクラスを持ち、従属オブジェクトのないオブジェクトを、ローカル eDirectory データベースからすべて削除します。このオプションによって、不明オブジェクトが削除対象としてマークされます。削除は eDirectory ツリーの他のレプリカと同時に、後で行われます。

**警告:** このオプションは、ConsoleOne または iManager を使用してオブジェクトの変更や削除ができない場合にのみ使用してください。

## [このデータベースに認識されているサーバ] のオプション

サーバでは次の修復オプションが使用できます。

- ◆ すべてのネットワークアドレスの修復

ローカル eDirectory データベース内で、すべてのサーバのネットワークアドレスをチェックします。このオプションでは、使用可能なトランスポートプロトコルに応じて、各サーバ名の SLP ディレクトリエージェントが検索されます。次に、各アドレスが、サーバオブジェクトのネットワークアドレスプロパティ、およびすべてのパーティション Tree オブジェクトの各レプリカプロパティのアドレスレコードと比較されます。アドレスが異なる場合は、同じになるように更新されます。

- ◆ 選択したサーバのネットワークアドレスの修復

ローカル eDirectory データベースファイル内にある特定のサーバのネットワークアドレスをチェックします。このオプションでは、SLP ディレクトリエージェントが、サーバ名に現在関連付けられているトランスポートプロトコルに応じて検索されます。

- ◆ サーバのフルネームを表示

サーバ名が長すぎてサーバテーブル内にすべてを表示できない場合に、完全なサーバ名を表示するのに使用します。このオプションは、-P オプションと同じです。詳細については、[521 ページの「-P」](#)を参照してください。

## 例

標準修復を実行し、/root/ndsrepair.log ファイルにイベントを記録する場合 ( またはログファイルがすでに存在していればそのログファイルに追加してイベントを記録する場合 ) は、次のコマンドを入力します。

```
ndsrepair -U -A no -F /root/ndsrepair.log
```

すべてのグローバルスキーマ操作とその詳細設定オプションのリストを表示するには、次のコマンドを入力します。

```
ndsrepair -S -Ad
```

データベースを強制ロックしてローカルデータベースを修復するには、次のコマンドを入力します。

```
ndsrepair -R -l yes
```

**注:** ndsrepair コマンドの入力内容は、オプションファイルによってリダイレクトできます。オプションファイルは、レプリカおよびパーティション操作に関連するオプションやサブオプションを含むテキストファイルです。これらはサーバに対する認証を必要としません。各オプションまたはサブオプションは改行によって区切られます。ファイルの内容が、適切な順序で指定されていることを確認します。適切な順序になっていないと、予期しない結果が発生する場合があります。

## ndsrepair のトラブルシューティング

### エラー - 786 ndsrepair の実行中

ndsrepair の実行中は、コンピュータの特定のパーティション内に DIB 空き領域の 3 倍のサイズが必要です。

## ndstrace の使用

ndstrace ユーティリティは、次の 3 つの主要部分で構成されています。

- ◆ 528 ページの「基本機能」
- ◆ 529 ページの「デバッグメッセージ」
- ◆ 531 ページの「バックグラウンド処理」

### 基本機能

ndstrace の基本機能は次のとおりです。

- ◆ Linux、Solaris、AIX、または HP-UX で ndstrace 画面のステータスを表示します。
- ◆ 一部の同期処理を開始します。

ndstrace 画面を起動するには、サーバプロンプトで次のコマンドを入力します。

```
/usr/bin/ndstrace
```

ndstrace の基本機能を開始するには、次の構文を使用してサーバプロンプトでコマンドを入力します。

```
ndstrace command_option
```

次の表では、入力可能なコマンドオプションのリストを示します。

オプション	説明
ON	基本トレースメッセージを含む eDirectory トレース画面を起動します。
OFF	トレース画面を無効にします。
ALL	eDirectory トレース画面を起動し、すべてのトレースメッセージを表示します。
AGENT	ON、BACKLINK、DSAGENT、JANITOR、RESNAME、および VCLIENT フラグと同等のトレースメッセージを含む eDirectory トレース画面を開始します。
DEBUG	デバッグに通常使用する定義済みのトレースメッセージのセットを有効にします。設定されるフラグは、ON、BACKLINK、ERRORS、EMU、FRAGGER、INIT、INSPECTOR、JANITOR、LIMBER、MISC、PART、RECMAN、REPAIR、SCHEMA、SKULKER、STREAMS、および VCLIENT です。
NODEBUG	トレース画面は使用可能なままで、以前に設定したデバッグメッセージはすべて無効にします。このオプションでは、メッセージも ON のコマンドオプションが設定された状態のままになります。

## デバッグメッセージ

ndstrace 画面が使用可能な場合、デフォルトのフィルタの設定に基づいて情報が表示されます。デフォルトで表示される情報の内容を変更するには、デバッグメッセージフラグを使用してフィルタを操作します。デバッグメッセージにより、eDirectory のステータスを確認し、問題が発生していないかどうかを検証できます。

各 eDirectory 処理には、デバッグメッセージのセットが含まれています。個々の処理中にそのデバッグメッセージを表示するには、プラス記号 (+)、および該当する処理名またはオプションを使用します。処理を表示しない場合は、マイナス記号 (-)、および該当する処理名またはオプションを使用します。次に例を示します。

メッセージ	説明
set ndstrace = +SYNC	同期メッセージを表示します。
set ndstrace = -SYNC	同期メッセージを非表示にします。
set ndstrace = +SCHEMA	スキーマメッセージを表示します。

また、ブール演算子の &(AND) および |(OR) を使用して、デバッグメッセージのフラグを結合することもできます。サーバコンソールでデバッグメッセージを制御する構文は、次のとおりです。

```
set ndstrace = +trace_flag [trace_flag]
```

または

```
set ndstrace = +trace_flag> [&trace_flag]
```

次の表では、デバッグメッセージ用のトレースフラグについて説明します。各トレースフラグは略語で入力できます。

トレースフラグ	説明
ABUF	eDirectory リクエストとの連携、または eDirectory リクエストへの応答として受信されたデータを含む、インバウンドおよびアウトバウンドパケットバッファに関するメッセージと情報です。
ALOC	メモリ割り当ての詳細について示すメッセージです。
AREQ	他のサーバまたはクライアントからのインバウンド要求に関するメッセージです。
AUTH	認証に関するメッセージとエラーレポートです。
BASE	最小限のデバッグレベルでのデバッグエラーメッセージ。
BLNK	バックリンクとインバウンドの破損通知メッセージおよびエラーレポートです。
CBUF	アウトバウンドDSクライアントリクエストに関するメッセージです。
CHNG	キャッシュ変更メッセージです。
COLL	以前に更新内容を受信したときのオブジェクトの更新情報に関するステータスおよびエラーレポートです。

トレースフラグ	説明
CONN	ローカルサーバが接続を試みている相手のサーバ、およびローカルサーバが接続できない原因となっている可能性のあるエラーとタイムアウトについての情報を示すメッセージです。
DNS	eDirectory 統合 DNS サーバプロセスに関するメッセージです。
DRLK	分散リファレンスリンクメッセージです。
DVRS	eDirectory が機能している可能性のある DirXML <sup>®</sup> ドライバ固有のエリアを示すメッセージです。
DXML	DirXML イベントの詳細について示すメッセージです。
FRAG	eDirectory メッセージを NCP サイズのメッセージに分解する、NCP <sup>™</sup> フラッグからのメッセージです。
IN	インバウンドの要求およびプロセスに関するメッセージです。
INIT	eDirectory の初期化に関するメッセージです。
INSP	ソースサーバのローカルデータベース内のオブジェクトの整合性に関するメッセージです。このフラグを使用すると、ソースサーバのディスクストレージシステム、メモリ、プロセッサの要求量が増加します。オブジェクトが破損しない限り、このフラグは有効に設定しないでください。
JNTR	ジャンタ、レプリカの同期、フラットクリーナなどのバックグラウンド処理に関するメッセージです。
LDAP	LDAP サーバに関するメッセージです。
LMBR	リンバ処理に関するメッセージです。
LOCK	ソースサーバのローカルデータベースロックの使用および操作に関するメッセージです。
LOST	消失エントリーに関するメッセージです。
MISC	eDirectory 内の異なるソースからのメッセージです。
MOVE	パーティションの移動操作、またはサブツリーの移動操作からのメッセージです。
NCPE	NCP レベルのリクエストを受信したサーバを示すメッセージです。
NMON	iMonitor に関するメッセージです。
OBIT	破損通知処理からのメッセージです。
PART	バックグラウンド処理およびリクエスト処理からのパーティション操作に関するメッセージです。
PURG	ページ処理に関するメッセージです。
RECM	ソースサーバのデータベースの操作に関するメッセージです。
RSLV	名前解決要求の処理に関するメッセージです。

トレースフラグ	説明
SADV	SLP (Service Location Protocol) のツリー名とパーティションの登録に関するメッセージです。
SCMA	スキーマの同期処理に関するメッセージです。
SCMD	スキーマ関連の操作の詳細について示すメッセージです。インバウンド同期とアウトバウンド同期の両方についての詳細を示します。
SKLK	レプリカの同期処理に関するメッセージです。
SPKT	eDirectory NCP サーバレベルの情報に関するメッセージです。
STRM	ストリーム構文の属性の処理に関するメッセージです。
SYDL	レプリケーション処理時の詳細について示すメッセージです。
SYNC	インバウンド同期トラフィック (サーバ側で受信される内容) についてのメッセージです。
TAGS	トレースオプションを識別するタグ文字列が表示されます。このトレースオプションでは、トレース処理で表示される各行のイベントが生成されます。
THRD	バックグラウンド処理 (スレッド) の開始時と終了時を示すメッセージです。
TIME	同期処理時に使用される遷移ベクトルに関するメッセージです。
TVEC	Synchronize Up To、レプリカ、および遷移ベクトルなどの属性に関するメッセージです。
VCLN	他のサーバへの接続の確立または切断に関するメッセージです。

ndstrace でデバッグメッセージを使用していると、特に便利なトレースフラグがあることが分かります。Novell サポートで多く使用されている ndstrace 設定には、次のようなショートカットがあります。

```
set ndstrace = A81164B91
```

この設定を使用すると、複数のデバッグメッセージを1つのグループとして使用できます。

## バックグラウンド処理

eDirectory のステータスを確認できるデバッグメッセージの他に、eDirectory バックグラウンド処理を強制的に実行するコマンドのセットも用意されています。バックグラウンド処理を強制的に実行するには、コマンドの先頭にアスタリスク (\*) を付けます。次に例を示します。

```
set ndstrace = *H
```

また、いくつかのバックグラウンド処理のステータス、タイミング、および制御を変更することもできます。これらの値を変更するには、コマンドの先頭に感嘆符 (!) を付けて新しいパラメータまたは値を入力します。次に例を示します。

```
set ndstrace = !H 15 (parameter_value_in_minutes)
```

eDirectory バックグラウンド処理を制御する各ステートメントの構文を次に示します。

```
set ndstrace = *trace_flag [parameter]
```

または

```
set ndstrace = !trace_flag [parameter]
```

次の表では、バックグラウンド処理のトレースフラグ、必要なパラメータ、およびトレースフラグが表示する処理のリストを示します。

トレースフラグ	パラメータ	説明
*A	None	ソースサーバのアドレスキャッシュをリセットします。
*AD	None	ソースサーバのアドレスキャッシュを無効にします。
*AE	None	ソースサーバのアドレスキャッシュを有効にします。
*B	None	ソースサーバ上で 1 秒後にバックリンク処理の実行を開始するようにスケジューリングします。
!B	時間	バックリンク処理の実行間隔を分単位で設定します。 デフォルトは 1500 分 (25 時間) です。 値の範囲は 2 ~ 10080 分 (168 時間) です。
*CT	None	ソースサーバのアウトバウンド接続テーブルと、テーブルの現在の統計情報を表示します。これらの統計情報には、他のサーバやクライアントからソースサーバへのインバウンド接続に関する情報は含まれていません。
*CTD	None	コンマ区切りの形式で、ソースサーバのアウトバウンド接続テーブルと、テーブルの現在の統計情報を表示します。これらの統計情報には、他のサーバやクライアントからソースサーバへのインバウンド接続に関する情報は含まれていません。
*D	レプリカ rootEntry ID	指定したローカルエントリ ID をソースサーバの [すべてのオブジェクトを送信] リストから削除します。エントリ ID では、サーバのローカルデータベースで固有のパーティションルートオブジェクトを指定する必要があります。通常、このコマンドは、サーバのアクセス不能が原因で Send All Updates 処理が何度試みられても失敗する場合にのみ使用します。
!D	時間	インバウンド同期およびアウトバウンド同期の間隔を指定した分数に設定します。 デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。
!DI	時間	インバウンド同期の間隔を指定した分数に設定します。 デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。



トレースフラグ	パラメータ	説明
!DO	時間	アウトバウンド同期の間隔を指定した分数に設定します。 デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。
*E	None	ソースサーバのエントリキャッシュを再初期化します。
!E	None	インバウンド同期およびアウトバウンド同期処理の実行を開始するようにスケジューリングします。
!EI	None	インバウンド同期処理の実行を開始するようにスケジューリングします。
!EO	None	アウトバウンド同期処理の実行を開始するようにスケジューリングします。
*F	None	ジャンタ処理の一部として、フラットクリーナ処理の実行がソースサーバ上で 5 秒後に開始されるようにスケジューリングします。
!F	時間	フラットクリーナ処理の実行間隔を分単位で設定します。 デフォルトは 240 分 (4 時間) です。 値の範囲は 2 ~ 10080 分 (168 時間) です。
*G	レプリカ rootEntry ID	指定したルートパーティション ID の変更キャッシュを再構築します。
*H	None	ソースサーバ上で直ちにレプリカ同期処理の実行を開始するようにスケジューリングします。
!H	時間	ハートビート同期処理の実行間隔を分単位で設定します。 デフォルトは 30 分です。 値の範囲は 2 ~ 1440 分 (24 時間) です。
*HR	None	メモリ内で最後に送信されたベクトルを消去します。
*I	レプリカ rootEntry ID	指定したローカルエントリ ID をソースサーバの [すべてのオブジェクトを送信] リストに追加します。エントリ ID では、サーバのローカルデータベースで固有のパーティションルートオブジェクトを指定する必要があります。レプリカの同期処理では、[すべてのオブジェクトを送信] リストがチェックされます。パーティションのルートオブジェクトのエントリ ID がリスト内に存在する場合、Synchronized Up To 属性の値に関係なく、eDirectory によってパーティション内のすべてのオブジェクトと属性が同期されます。
!!	時間	ハートビート同期処理の実行間隔を分単位で設定します。 デフォルトは 30 分です。 値の範囲は 2 ~ 1440 分 (24 時間) です。
*J	None	レプリカの同期処理の一部として、ソースサーバ上でパージ処理の実行を開始するようにスケジューリングします。

トレースフラグ	パラメータ	説明
!J	時間	<p>ジャニタ処理の実行間隔を分単位で設定します。</p> <p>デフォルトは 2 分です。 値の範囲は 1 ~ 10080 分 (168 時間) です。</p>
*L	None	<p>ソースサーバ上で 5 秒後にリンバ処理の実行を開始するようにスケジューリングします。</p>
*M	Bytes	<p>ソースサーバの ndstrace.log ファイルで使用する最大ファイルサイズを変更します。このコマンドは、デバッグファイルのステータスに関係なく使用できます。bytes の値は 10000 バイトから 100MB までの範囲で指定します。この範囲外の値が指定された場合、変更は発生しません。</p>
!M	None	<p>eDirectory で使用されるメモリの最大量をレポートします。</p>
!N	0 1	<p>名前の形式を設定します。</p> <p>0 の場合は 16 進数のみです。 1 の場合は完全ドット形式です。</p>
*P	None	<p>調整可能なパラメータとそのデフォルトの設定を表示します。</p>
*R	None	<p>TTF ファイルをリセットします。デフォルトでは sys:system\ndstrace.dbg ファイルです。このコマンドは、SET パラメータの NDS Trace File Length Set to Zero と同じ働きをします。</p>
*S	None	<p>サーバ上のレプリカを同期する必要があるかどうかをチェックするスカルク処理をスケジューリングします。</p>
!SI	時間	<p>インバウンドスキーマ同期処理の実行間隔を分単位で設定します。</p> <p>デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。</p>
!SO	時間	<p>アウトバウンドスキーマ同期処理の実行間隔を分単位で設定します。</p> <p>デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。</p>
!SIO	時間	<p>時間を分単位で指定し、その間のインバウンドスキーマ同期処理を無効にします。</p> <p>デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。</p>
!SO0	時間	<p>時間を分単位で指定し、その間のインバウンドスキーマ同期処理を無効にします。</p> <p>デフォルトは 24 分です。 値の範囲は 2 ~ 10080 分 (168 時間) です。</p>
*SS	None	<p>強制的に即時スキーマの同期を実行します。</p>

トレースフラグ	パラメータ	説明
*SSA	None	スキーマの同期処理の実行を即時に開始するようにスケジューリングします。過去 24 時間に同期が実行されていた場合でも、すべてのターゲットスキーマでスキーマの同期が強制的に実行されます。
*SSD	None	ソースサーバの [ターゲットスキーマ同期] リストをリセットします。このリストでは、スキーマの同期処理の実行中にソースサーバと同期する必要のあるサーバが識別されず、レプリカを保持していないサーバは、サーバオブジェクトとレプリカを保持しているサーバのターゲットリストに包含されるように要求を送信します。
*SSL	None	ターゲットサーバのスキーマ同期リストを印刷します。
*ST	None	ソースサーバ上のバックグラウンド処理のステータス情報を表示します。
*STX	None	ソースサーバ上のバックリンク処理 (外部参照) のステータス情報を表示します。
*STS	None	ソースサーバ上のスキーマ同期処理のステータス情報を表示します。
*STO	None	ソースサーバ上のバックリンク処理 (破損通知) のステータス情報を表示します。
*STL	None	ソースサーバ上のリンバ処理のステータス情報を表示します。
!T	時間	サーバの稼動状態のチェックの実行間隔を分単位で設定します。  デフォルトは 30 分です。 値の範囲は 1 ~ 720 分 (12 時間) です。
*U	サーバのオプションの ID	コマンドにエン트리 ID が含まれていない場合は、以前に「down」から「up」にラベルが付加された任意のサーバのステータスを変更します。コマンドにローカルエン트리 ID が含まれている場合は、指定されたサーバのステータスを「down」から「up」に変更します。エン트리 ID は、ソースサーバのデータベースで固有であり、サーバを表すオブジェクトを参照する必要があります。
!V	リスト	制限のある eDirectory パージョンのリストを表示します。パージョンが表示されない場合は制限がないことを示します。各パージョンはコンマで区切られます。
*Z	None	現在、スケジューリングされているタスクを表示します。

## ConsoleOne のトラブルシューティング

ConsoleOne のトラブルシューティングの詳細については、『*ConsoleOne ユーザガイド*』の「トラブルシューティング ([http://www.novell.com/documentation/japanese/consol13/c1\\_enu/data/hlgvum.html](http://www.novell.com/documentation/japanese/consol13/c1_enu/data/hlgvum.html))」を参照してください。

### eDirectory ツリーをブラウズできない

次のいずれかのに該当する場合は、eDirectory ツリーをブラウズできません。

- ◆ eDirectory ツリーの Tree オブジェクトの削除、名前変更、移動が行われた場合。  
ネットワークからログアウトして、変更を表示するために再度ツリーにログインした場合。
- ◆ ndsd デーモンが停止している場合。  
次のコマンドのいずれかを入力し、デーモンを再起動します。
  - ◆ Linux の場合 `:/etc/rc.d/init.d/ndsd start`
  - ◆ Solaris の場合 `:/etc/init.d/ndsd start`
  - ◆ AIX の場合 `:/etc/ndsd start`
  - ◆ HP-UX の場合 `:/sbin/init.d/ndsd start`
- ◆ 次のエラーメッセージが表示される場合。

An attempt to resolve SVC (switched virtual circuit) failed.

これは、ツリーがプライマリツリーであるのに対して、サーバがプライマリサーバではないためです。サーバをプライマリサーバに設定してください。

これらの操作のいずれかを実行する場合は、ツリーをブラウズするために新しい ConsoleOne ウィンドウを開いてください。

### NetWare サーバ上での純粋な IP 環境の実行

NetWare サーバ上の純粋な IP 環境では、ConsoleOne が実行されているサーバの eDirectory ツリーは ConsoleOne で認識されません。

これを修正するには、`autoexec.ncf` ファイルに次の変更を加えてください。

- ◆ TCPIP の LOAD および BIND ステートメントの後に `LOAD SCMD.NLM` というステートメントを追加します。
- ◆ サーバ ID に `IPX™` を追加します。  
サーバを再起動します。

この問題の詳細については、Novell Knowledgebase の [Solution #2943528](http://support.novell.com) (<http://support.novell.com>) を参照してください。

# 破損通知

ディレクトリに保存される破損通知については非常に複雑なため、ビジネスでうまく利用されていない場合があります。いくつかのディレクトリ製品とは異なり、Novell eDirectory ではオブジェクト間の参照整合性が確保されています。たとえば、グループ A にユーザ B というメンバが登録されている場合、ユーザ B が削除されるとディレクトリでは自動的にグループ A からユーザ B への参照が削除されます。破損通知は、eDirectory によってオペレーショナル属性としてオブジェクトに保存されます。これにより、削除、移動、名前の変更、復元、およびその他の操作中に、二重に参照整合性が確保されます。

破損通知には大きく分類して 3 つの種類があります。

- ◆ プライマリ破損通知には、停止 (0001)、復元 (0000)、移動 (0002)、新規 RDN(0005)、およびツリーの新規 RDN(0008) の各種類があります。
- ◆ セカンダリ破損通知は、一般的にプライマリ破損通知に関連付けられており、プライマリ破損通知で指定された操作の通知が必要なエージェントおよびパーティションを表します。セカンダリ破損通知には、バックリンク (0006)、使用中 (000C)、およびツリーの移動 (000a) の各種類があります。
- ◆ トラッキング破損通知には移動禁止 (0003)、古い RDN(0004)、およびツリーの古い RDN(0007) の各種類があります。

トラッキング破損通知以外の破損通知は、次の同期ステータスのセットを使用して移動する必要があります。

- ◆ 初期化ステータスまたは発行済み (0)
- ◆ 通知済み (1)
- ◆ パージ準備完了 (2)
- ◆ パージ可能 (4)

ステータスは破損通知属性のフラグフィールドで記録されます。破損通知が次のステータスに進む前に、現在のステータスは必ず実オブジェクトのすべてのレプリカに同期されます。リング内のすべてのレプリカが破損通知ステータスを与えられているかどうかを判断するために、遷移ベクトルからベクトルが計算されます。eDirectory 8.6 以降では、保存されていない破損通知ベクトルが使用されます。以前のバージョンの eDirectory では、パージベクトルが使用されます。破損通知の変更タイムスタンプ (MTS) が破損ベクトルよりも古い場合、担当サーバは該当する破損通知を次のステータスに進めることができます。

「バックリンク」のセカンダリ破損通知の場合、該当する破損通知を含むオブジェクトのマスタレプリカを持つエージェントがステータスを進めます。「使用中」のセカンダリ破損通知の場合、レプリカが存在している間は該当する破損通知を作成したエージェントがステータスを進めます。レプリカが存在しない場合、パーティションのマスタを保持しているエージェントが「使用中」の破損通知のステータスを進めます。「ツリーの移動」の破損通知の場合、ルートパーティションのマスタがステータスを進めます。

プライマリ破損通知は、すべてのセカンダリ破損通知が最後のステータスまで進められた後でのみ、ステータスを進めることができます。プライマリ破損通知が最後のステータスまで進んだ後で、そのステータスがリング内のすべてのサーバに同期されると、残っているのは属性を持たないオブジェクトであるオブジェクトハスクのみとなり、これらはシステムのパージプロセスによってパージされます。トラッキング破損通知は、プライマリ破損通知の削除の準備が完了した後か、Inhibit\_move の場合はプライマリ破損通知がマスタレプリカの OBF\_NOTIFIED ステータスに移動された後で削除されます。

破損通知の処理を担当するレプリカは、指定したパーティションがインバウンド同期サイクルを終了した後で、パーティションごとにスケジュールされているバックグラウンド処理 (破損通知処理) を実行します。パーティションにその他のレプリカがない場合、アウトバウンドレプリケーション処理がハートビート間隔でスケジュールされたままになります。その後、アウトバウンドレプリケーション処理によって破損通知処理が開始されます。破損通知処理は手動ではスケジュールできず、また、その必要もありません。同期化が実行されると、遷移ベクトルが更新され、ページベクトルおよび Obit ベクトルを進めます。これらのベクトルが進められると、破損通知のステータスを進めることができます。これと同時に、インバウンド同期に自動スケジュールリングが実行されると、破損通知処理サイクルが完了します。すなわち、破損通知処理の起動要因はオブジェクト同期です。

削除されたオブジェクトの場合、「停止」のプライマリ破損通知に関連するすべての破損通知が最後のステータス (ページ可能) まで進められ、そのステータスがすべてのレプリカに同期された後で、新しい処理がデータベースに残っているエントリハスクの削除を担当します。これらのハスクを削除するために、ページ処理が自動的に実行されます。ページ処理のスケジュールおよび自動スケジュール間隔の調整は、iMonitor の [エージェント環境設定](#) ページを使用して手動で設定することができます。

## 例

このセクションでは、次の例を紹介します。

- ◆ [538 ページ](#)の「オブジェクトの削除」
- ◆ [539 ページ](#)の「オブジェクトの移動」

### オブジェクトの削除

#### 1 プライマリ破損通知 OBT\_DEAD を追加します。

バックリンクの属性には、このオブジェクトに関連し、このエントリに対する変更を通知する必要があるサーバのリストが含まれています。バックリンクの属性のリストに含まれる各 DN およびエントリのパーティションレプリカ属性のリストに含まれるすべてのサーバに対して、eDirectory はバックリンク破損通知を追加します。プライマリ破損通知 (OBT\_DEAD) の作成時刻は、セカンダリ破損通知に保存されます。

使用中の属性には、このオブジェクトに関連し、このエントリに対する変更を通知する必要があるパーティションのリストが含まれています。使用中の属性のリストに含まれているすべての DN に対して、eDirectory は使用中の破損通知を追加します。プライマリ破損通知 (OBT\_DEAD) の作成時刻は、セカンダリ破損通知に保存されます。

#### 2 破損通知以外のすべての属性を削除します。

次に、アウトバウンドレプリケーションプロセスによって、レプリカリング内にある他のすべてのサーバに変更が同期されます。

このパーティションの次のインバウンド同期が実行されるたびに、破損通知処理が開始され、次の処理が実行されます。

- ◆ 最小遷移ベクトルである時間ベクトルを計算し、ページベクトルとして参照されます。比較的新しいバージョンの eDirectory では、2 番目に小さいベクトルが計算されます。このベクトルは、サブオーディネートリファレンスのレプリカではなく、破損通知ベクトルと呼ばれます。

- ◆ このパーティション内にあるそれぞれの破損通知が検査されます。

該当する破損通知がプライマリ破損通知で、セカンダリ破損通知がなく、破損通知の属性変更タイムスタンプ (MTS) がページベクトルよりも古い場合、すべてのサーバが変更を確認済みとしてこの破損通知は削除されます。

該当する破損通知がバックリンク破損通知で、このサーバがマスタの場合、このサーバが破損通知の処理を担当します。

**重要:** ステータスが完了していない場合、このステータスに必要な操作を実行します。これは外部参照を通知するときにも最も頻繁に実行されます。

該当する破損通知が使用中の破損通知で、(破損通知の MTS のレプリカ番号とローカルのレプリカ番号の比較から判断して) このサーバで削除が発生している場合、このサーバがこの破損通知の処理を担当します。

- ◆ このサーバが特定のセカンダリ破損通知のタイプ (バックリンクまたは使用中) の処理を担当し、1 つのエントリ内にある該当するタイプのすべてのセカンダリ破損通知が同じステータスであり、該当するステータスに必要な処理 (例えば、サーバへの通知) がすべての破損通知で完了していて、さらにその破損通知のタイプの MTS が破損通知ベクトルよりも古い場合、該当するタイプのすべてのセカンダリ破損通知を次のステータスに進めることができます。

## オブジェクトの移動

移動は**削除**と非常によく似ていますが、次に挙げる操作が違います。

- ◆ プライマリ破損通知が移動元に配置される前に、エントリの一部が移動先のコンテナに作成され、そのエントリの一部にトラッキング破損通知 (OBT\_INHIBIT\_MOVE) が配置されます。このトラッキング破損通知は、エントリが移動元から完全に移動される前にエントリが移動されたり、パーティション操作に加わるのを防ぐために配置されます。
- ◆ ソースエントリでは、プライマリ破損通知は OBT\_MOVED です。
- ◆ プライマリ破損通知 (OBT\_MOVED) のステータスが通知済み (ソースのすべてのレプリカにエントリが移動されたことを通知した状態) になり、すべての外部参照に通知が完了すると、トラッキング破損通知 (OBT\_INHIBIT\_MOVE) が移動先エントリから削除されます。

## 停止および孤立した破損通知の影響

破損通知を含むオブジェクトは、エージェントのアウトバウンド同期の実行時、およびインバウンド同期サイクルの最後に実行されるようにスケジュールされている破損通知処理の実行時に調査の対象となります。

## 予防策

定期的に iMonitor サーバ情報レポートを実行してください。このレポートは、ツリー全体を調べて、検索可能な各 NCP サーバと通信し、検知したすべてのエラーをレポートします。このレポートを使用して、時刻同期およびリンバの問題を診断できます。また、現在のサーバ自体が他のすべてのサーバと通信可能であると認識しているのかも知ることができます。環境設定ページで選択されている場合、サーバはツリー内にある各サーバの NDS エージェントヘルス情報を生成することもできます。サーバ情報レポートの実行の詳細については、[189 ページの「レポートの設定と表示」](#)を参照してください。

iMonitor2.0 以降のバージョンを使用する場合は、[Errors and Health sub-report] のレポートオプションが有効になっていることを確認してください。レポートでは次の項目が確認されます。レポートを参照し、エラーがないことを確認してください。

- ◆ iMonitor に保存されている ndsimonhealth 環境設定ファイルの情報に基づき (177 ページの「環境設定ファイル」を参照)、ツリー全体で正しいディレクトリパッチを実行していることを確認するために eDirectory エージェントのバージョンをチェックする。
- ◆ すべてのサーバが Timesync の許容範囲内にある。
- ◆ このサーバが他のすべてのサーバと通信できる。
- ◆ このツリーから不適切または不完全に削除されたサーバがない。
- ◆ ヘルスサブレポートで、レプリケーション同期時刻の許容範囲外にパーティションがあるかどうかを確認する。

iMonitor1.5 を使用する場合はエラーレポートオプションを選択します。レポートでは次の項目が確認されます。レポートを参照し、エラーがないことを確認してください。

- ◆ エージェントのバージョンが表示される。ツリー内のすべてのサーバで、Novell Support の Web サイト (<http://support.novell.com>) から入手できる最新の eDirectory Support Pack が実行されている。
- ◆ すべてのサーバが Timesync の許容範囲内にある。
- ◆ このサーバが他のすべてのサーバと通信できる。
- ◆ このツリーから不適切または不完全に削除されたサーバがない。

iMonitor 破損通知リスティングレポートまたは iMonitor オブジェクト統計情報レポートを使用して、システムにある破損通知を検索できます。処理が実行されていないと思われる破損通知を見つけた場合は、540 ページの「トラブルシューティングのヒント」を参照してください。

## トラブルシューティングのヒント

破損通知が処理されない場合、大きく分けて次の 2 つの原因が考えられます。それは、破損通知が孤立している場合 (破損通知がすべてのサーバではなく一部のサーバにのみ存在する場合)、または破損通知が停止している場合 (破損通知がすべてのサーバに存在するが、ステータスが何らかの理由で進まない場合) です。

次の項目を参照して、孤立または停止した破損通知の問題を解決してください。

- あわてずに対応してください。
- オブジェクトの破損通知がこのサーバに保存されていない (該当するオブジェクトが外部参照の) 場合、次を実行します。
  - ◆ 一致する破損通知が実オブジェクトに含まれているかどうかをチェックします。一致する破損通知が含まれていない場合、この破損通知は孤立しています。詳細については、542 ページの「外部参照の孤立した破損通知の解決」を参照してください。
  - ◆ 一致する破損通知が実オブジェクトに含まれている場合、実オブジェクトの破損通知の問題を解決してから、外部参照にある破損通知の問題を特定してください。



- 破損通知が正確に同期されていることを確認してください。
  - ◆ iMonitor の [\[エージェント同期\]](#) ページを使用して同期エラーをチェックおよび解決してください。
  - ◆ 破損通知は、レプリカリングのコピーを保持するすべてのエージェントがステータスの変更を確認した後でのみステータスを変更できます。すべてのレプリカがデータを認識したことを確認するには、次のようないくつかの方法があります。

破損通知を含むエントリを参照しながら、エントリ同期リンクをクリックします。すべてのレプリカに同期されていない属性がすべて表示されます。

破損通知属性値の中で、最も古いタイムスタンプを検索します。検索されたタイムスタンプの時刻と現在時刻の差が、[\[パーティション同期\]](#) ページの最大リングデルタフィールドの間隔よりも大きい必要があります。

遷移ベクトルを調査します。
- すべてのサーバ通信が機能していることを確認するには、iMonitor の [サーバ情報レポート](#) を実行します。
- エラーを検索するには、[エージェントプロセスステータス:破損通知](#) を検索してください。
  - ◆ エージェントプロセスステータス:破損通知で起こりうる一般的な問題には、次のものがあります。
    - 625、-622、-634、および -635 の通信エラー。詳細については、「[サーバ情報レポート](#)」を参照してください。
    - 601 および -603 は、適切に削除されていないサーバ、またはサーバオブジェクトに不明なベースクラスが含まれるサーバを示します。
  - ◆ このページに表示されるエラーは致命的なものではありません。そのパーティションで破損通知処理が次回実行されるときに、この操作が再試行されます。このページに表示された問題を解決して、再試行まで待機してください。
- 破損通知オブジェクトを表示した状態で、リング周辺の破損通知を比較しながらレプリカリングを調べます。
  - ◆ 破損通知のコピーがないレプリカがあり、すべての属性値がパーシステントでない場合、このオブジェクトはレプリカリング周辺で矛盾しており、破損通知が孤立していると考えられます。詳細については、[542 ページの「孤立した破損通知の解決」](#)を参照してください。
  - ◆ オブジェクトがすべてのレプリカに矛盾なく存在している場合、同期エラー、または破損通知処理にエラーが発生しているために次のステータスに進まない可能性があります。
- 必要に応じて [\[Trace with the Obituary\]](#) オプションを有効にして、破損通知処理の詳細を検査してください。
- 今後、破損通知の問題を回避するには、eDirectory 8.6 の最新のサポート パックにアップグレードしてください。破損通知についての既知の問題がすべて修正されています。

## 解決方法

540 ページの「トラブルシューティングのヒント」を参照して、適切な解決方法を使用してください。

これらの解決方法を使用する前に、データが安全であることを確認してください。ディレクトリデータベースファイル、サーバ環境設定、およびトラスティのバックアップが必要となる場合があります。成功率を高め、今後生じる問題を最小限に抑えるためには、最新の eDirectory サポートパックにアップグレードしてください。

### 孤立した破損通知の解決

- ◆ **推奨される方法**：レプリカリング内のサーバのいずれかに eDirectory 8.6 以降のバージョンを使用する場合、iMonitor のオブジェクトを参照し、[Send Single Entry( 単一エントリを送信 )] を選択します。これにより、その他のすべてのレプリカに非委任の送信が実行されます。
- ◆ **避けるべき方法**：孤立した破損通知のコピーを持つレプリカリングにあるすべてのサーバが eDirectory 8.6 よりも前のバージョンの場合に、DSBrowse を -a オプションでロードして、オブジェクトを参照し、エントリにタイムスタンプを設定します。これにより、このサーバに存在するオブジェクトがそのまま委任されたコピーとなります。Novell では、実際にはオブジェクトを委任されたものと設定することはお勧めしません。

### 外部参照の孤立した破損通知の解決

- ◆ **推奨されない方法**：タイムスタンプオプションが選択されている DSRepair を実行します。
- ◆ **推奨されない方法**：実レプリカをサーバに移動し、使用可能な状態になってから破損通知が処理されるのを待ちます。破損通知が処理されない場合は、540 ページの「トラブルシューティングのヒント」にある情報を参照して実レプリカに移動されたオブジェクトの問題を解決してください。破損通知が処理されたら、レプリカは削除してもかまいません。

## 以前の操作

過去には、停止した破損通知を解決するためにいくつかの異なる手段をとりました。これらの手段の一部は、費用のかかるパーティション操作、または将来的に問題の原因となる可能性のあるドキュメント化されていない機能の使用に関連しています。

1 つ目に使用されていたのは、マスタを保持しているレプリカを切り替える方法です。マスタはさまざまなステータスを通じてバックリンクの破損通知の移動を担当しているエージェントなので、この手段が有効な場合もあります。レプリカに矛盾がありマスタが削除されたオブジェクトを保持していない場合、該当するマスタを、破損通知と一緒に削除されたエントリを保持していたエージェントに切り替えると、新規エージェントに破損通知のステータスを次に進めてページすることのできるライセンスが与えられます。[Send Single Entry( 単一エントリを送信 )] を選択すると、より確実に危険性の低い方法で、レプリカの矛盾が原因で停止している破損通知の問題を解決します。

2 つ目に使用されていたのは、DSRepair を実行してすべての破損通知を削除するスイッチを使用する方法です (DSRepair を起動して、停止している破損通知を解決するサードパーティ製のアプリケーションがあります)。この方法はお勧めできません。これらのスイッチを使用すると、このエージェントにあるすべての破損通知が削除されます。したがって停止していない破損通知も削除される可能性があり、レプリカの矛盾性がさらに深まり、より多くの停止している破損通知が作成される場合があります。これは分散された操作ではないので、停止している破損通知のあるすべてのサーバで DSRepair を

実行する必要があります。この操作は、これらのサーバの1つで処理未了のまま削除されてしまうパーティションの破損通知を取得する可能性を高めます。処理未了のまま破損通知を削除すると、孤立した破損通知を新たに生み出す可能性があり、その結果として数年後にレプリカタイプの変更、新規レプリカの追加、またはその他のパーティション操作を実行したときに問題が生じる場合があります。

3 つ目に使用されていたのは、カスタムモード操作で DSBrowse を使用してエントリにタイムスタンプを設定するか、または RSRepair で -OT スイッチを使用してオブジェクトを委任されたものと設定する方法です。この操作によりエントリは委任されたものと設定され、他のすべてのレプリカにこのエントリを同期します。その他のサーバで変更されたデータを失う可能性があるため、この操作の実行には細心の注意を払う必要があります。破損通知のクリーンアップの方法としては、なるべく使用しないことをお勧めします。

## DS がロードされない場合の HTTPSTK へのアクセス

DS がロードされない場合に HTTPSTK(HTTP プロトコルスタック) にアクセスできる管理者ユーザを事前に設定できます。事前に設定された管理者ユーザ (SAdmin) には、eDirectory 管理者ユーザオブジェクトと同等の権利があります。サーバが、eDirectory が適切に機能していない状態の場合、このユーザとしてサーバにログインし、eDirectory を使用せずに実行できる必要なすべての診断およびデバッグ作業を実行します。

- ◆ 543 ページの「NetWare で SAdmin パスワードを設定する」
- ◆ 544 ページの「Windows で SAdmin パスワードを設定する」
- ◆ 544 ページの「Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する」

## NetWare で SAdmin パスワードを設定する

NetWare リモートマネージャを使用して SAdmin ユーザオブジェクトを有効にし、このオブジェクトのパスワードを設定または変更します。SAdmin のパスワードを設定または変更するには、eDirectory サーバ上で HTTPSTK.NLM が実行されてる必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。


**http://server's TCP/IP address:port**

次に例を示します。

http://137.65.123.11:8008

注：デフォルトの代替ポート番号は 8008 です。NetWare リモートマネージャの環境設定ページでこの値を変更した場合は、その新しいポート番号を入力します。

サーバ名と IP アドレスの解決のために DNS(Domain Name Service) をネットワークにインストールしてある場合は、IP アドレスの代わりにサーバの DNS 名を入力することもできます。

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 [環境設定] ボタン  [Enable Emergency Account(SADMIN User)] の順にクリックし、[パスワードの設定] をクリックします。
- 5 SAdmin のパスワードを指定してから、入力したパスワードを確認します。
- 6 [設定] をクリックします。

## Windows で SAdmin パスワードを設定する

DHOST リモートマネージャページ (/dhost URL またはルートページからアクセス可能) を使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで dhost.exe を実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 HTTP サーバをクリックしてから、SAdmin パスワードを指定します。
- 5 指定したパスワードを確認入力して、[送信] をクリックします。

## Linux、Solaris、AIX、および HP-UX で SAdmin パスワードを設定する

SAdmin のパスワードの設定には、DHOST リモート管理ページまたは ndsconfig ユーティリティを使用できます。

### DHOST リモート管理ページ

DHOST リモートマネージャページ (/dhost URL またはルートページからアクセス可能) を使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで Novell eDirectory サーバを実行している必要があります。

- 1 Web ブラウザを開きます。
- 2 アドレス (URL) フィールドに、次の形式で入力します。

**http://server.name:port/dhost**

たとえば、次のように入力します。

http://MyServer:80/dhost

DHost iConsole へのアクセスに、サーバの IP アドレスを使用することもできます。次に例を示します。

http://137.65.135.150:80/dhost

- 3 ユーザ名、コンテキスト、パスワードを指定します。
- 4 HTTP サーバをクリックしてから、SAdmin パスワードを指定します。
- 5 指定したパスワードを確認入力して、[送信] をクリックします。

### ndsconfig

ndsconfig ユーティリティを使用して、SAdmin パスワードを設定します。SAdmin パスワードの設定や変更を行うには、eDirectory サーバで ndsd を実行している必要があります。

サーバコンソールから、次のように入力します。

```
ndsconfig set http.server.sadmin-pwd=password
```

ここで *password* は、新しい SAdmin パスワードです。

ndsconfig ユーティリティの使用の詳細については、『Novell eDirectory 8.7.3 インストールガイド』の「**ndsconfig ユーティリティパラメータ**」を参照してください。

## SNMP のトラブルシューティング

このセクションには、すべてのプラットフォーム上の SNMP のトラブルシューティングについての情報が含まれています。

- ◆ 545 ページの「必要なトラップが生成されない」
- ◆ 545 ページの「Linux 上の SNMP」
- ◆ 546 ページの「HP-UX 上の SNMP」

### 必要なトラップが生成されない

トラップは、対応するバーブの要求がサーバで受信された場合にのみ送信されます。それ以外の場合にトラップは送信されません。たとえば、ndsRemoveEntry (トラップ番号 108) の要求が送信された場合にのみ、ndsDeleteAttribute が送信されます。ただし、アプリケーションでは常に ACL が読み込まれ、そのユーザに削除操作を実行する十分な権利があるかどうかを確認されます。この場合、ndsDeleteAttribute トラップは生成されません。ただし、iMonitor を使用して特定のサーバにバーブ統計情報を表示できます。

すべてのイベントの発生時にトラップを取得するには、時間間隔を 0 に設定します。

トラップを有効にすると、失敗した場合にのみ送信できます。トラップを有効にすると、すべての状態でトラップを取得できます。

### Linux 上の SNMP

```
ndssnmppsa: 共有ライブラリのロード中のエラー: libucdmibs-0.4.2.1.so: 共有オブジェクト  
ファイルを開けません: 該当するファイルまたはディレクトリはありません
```

または

```
ndssnmppsa: 共有ライブラリのロード中のエラー: libucdagent-0.4.2.1.so: 共有オブジェ  
クトファイルを開けません: 該当するファイルまたはディレクトリはありません
```

または

```
ndssnmppsa: 共有ライブラリのロード中のエラー: libsnmp-0.4.2.1.so: 共有オブジェ  
クトファイルを開けません: 該当するファイルまたはディレクトリはありません
```

システム上に不正なバージョンの ucd-snmp が存在しているか、または必要なライブラリへのリンクが存在しない可能性があります。

この問題を解決するには、ucd-snmp-4.2.1、ucd-snmp-4.2.2、または ucd-snmp-4.2.3 をインストールします。

たとえば、システムに ucd version 4.2 がインストールされている場合、次のような ucd バージョン 4.2.1 へのリンクを作成する必要があります。

```
ln -s /usr/lib/libucdagent.so.4.2 /usr/lib/libucdagent-0.4.2.1.so
ln -s /usr/lib/libsnmp.so.4.2 /usr/lib/libsnmp-0.4.2.1.so
ln -s /usr/lib/libucdmibs.so.0.4.2 /usr/lib/libucdmibs-0.4.2.1.so
```

## HP-UX 上の SNMP

### MIB ブラウザから SNMP マスタエージェントへの接続中のエラー

NIB ブラウザから SNMP マスタエージェントへの接続中にエラー(タイムアウトエラーなど)が発生した場合、次を実行します。

- ◆ SNMP マスタエージェントが起動され、実行中であることを確認します。

これは、次に示す `ps` コマンドを使用して確認できます。

```
ps -ef | grep snmpdm
```

- ◆ `/var/adm/snmpd.log` ファイルでエラーおよび警告メッセージを参照します。

### NET-SNMP-5.0.8 の環境設定の問題

`/usr/adm/snmpd.log` ファイルで、NET-SNMP-5.0.8 マスタエージェントに関するエラーおよび警告メッセージを参照します。

注: NET-SNMP マスタエージェントを停止し、再起動する場合は、`ndssnmpsa` も再起動する必要があります。

### NAA エージェントの環境設定の問題

`/var/adm/snmpd.log` ファイルで、NAA エージェントに関するエラーおよび警告メッセージを参照します。

ログメッセージを有効にした状態で NAA エージェントを起動してください。次を実行して `naaagt` を起動します。

```
/usr/sbin/naaagt -m APALL
```

注: デフォルトでは、(-k オプションで `naaagt` が起動されていない限り) `snmpdm` が終了した時点で `naaagt` は自動的に終了します。詳細については、`naaagt` のマニュアルページを参照してください。

### MIB ブラウザから SNMP 検索要求を取得できない

NET-SNMP-5.0.8 が設定、起動、および実行されていることを確認します。

eDirectory SNMP サブエージェント (`ndssnmpsa`) の環境設定の問題については、`/var/nds/ndssnmpsa.log` ファイルを参照してください。デバッグメッセージを取得するには、次のように `ndssnmpsa` を冗長モードで起動します。

```
/usr/bin/ndssnmpsa -v 3 -l 3
```

`v` の部分は冗長モード、`l` の部分はログモードを示します。

### トラップが SNMP コンソールまたは MIB ブラウザで受信されない

`net-snmp` 環境設定にトラップの送信先が入力されていることを確認します。

設定方法の詳細については、[350 ページの「NET-SNMP マスタエージェントの起動 / 設定」](#)のセクションを参照してください。

## コンテナ管理者の eDirectory インストールの失敗

eDirectory 8.7.3 のインストールプログラムでは、サーバが格納されているコンテナのスーパーバイザ権を持つ管理者によるインストールがサポートされています。これを実行するには、eDirectory をインストールする最初のサーバにスキーマを拡張する [Root] のスーパーバイザ権がある必要があります。この点から、後続のサーバには [Root] のスーパーバイザ権は必要ありません。ただし、eDirectory 8.7.3 では、最初に eDirectory 8.7.3 がインストールされているプラットフォームによっては一部のスキーマが拡張されない場合があります、以降異なるプラットフォームでサーバをインストールするときに [Root] のスーパーバイザ権が要求される場合があります。

eDirectory 8.7.3 を複数のプラットフォームにインストールする場合は、各プラットフォームでインストールする最初のサーバの [Root] のスーパーバイザ権があることを確認してください。たとえば、eDirectory 8.7.3 をインストールする最初のサーバが NetWare で実行されていて、eDirectory 8.7.3 を Solaris でもインストールする場合、各プラットフォームの最初のサーバは [Root] のスーパーバイザ権を持っている必要があります。それ以降、各プラットフォームでインストールする場合は、サーバがインストールされているコンテナに対するコンテナ管理者の権利のみが必要になります。

追加情報については、Novell eDirectory 8.7.x Readme Addendum のソリューション [NOVL81742 \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/10073723.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/10073723.htm) を参照してください。

## Sun One スキーマの Novell eDirectory への移行

Sun One スキーマを Novell eDirectory に移行するには、次の手順に従ってください。

[547 ページの「手順 1: スキーマキャッシュの更新操作を実行する」](#)

[548 ページの「手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する」](#)

[550 ページの「手順 3: LDIF ファイルのインポート」](#)

### 手順 1: スキーマキャッシュの更新操作を実行する

次のコマンドを使用して、スキーマの比較中に検出されたエラーをエラーファイルに書き込むことができます。

```
ice -e LDIF error file name -C -a -SLDAP -s Sun ONE server -p Sun ONE port -DLdap -s eDirectory server -p eDirectory port
```

次に例を示します。

```
ice -e err.ldf -C -a -SLDAP -s sun_srv1 -p sun_port1 -DLdap -s edir_srv2 -p edir_port2
```

スキーマの比較中に検出されたエラーはすべてエラーファイル (例では `err.ldf`) に書き込まれます。Root DSE を読み込むためにサーバから認証が要求されない限り、この操作を実行するためにログインする必要はありません。Microsoft アクティブディレクトリでは、Root DSE を読み込むための認証が要求されます。

## 手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する

- ◆ Sun ONE では、eDirectory では定義されていないいくつかのスキーマ定義が公式に定義されています。これらの定義には、「objectClasses」、「attributeTypes」、「ldapSyntaxes」および「subschemSubentry」などの属性が含まれます。これらの定義は内部に存在し、スキーマにとって非常に重要です。したがって、これらの定義を変更することはできません。これらの定義を変更しようとする、次のエラーが発生します。

LDAP エラー : 53 ( DSA が動作しません )

これらの定義の参照が含まれるすべてのレコードに対して、次のエラーが表示されます。

LDAP エラー : 16 : ( 該当する属性はありません )

したがって、これらのオブジェクトへの参照が含まれるレコードまたはこれらの定義の変更を試行するレコードは、LDIF エラーファイル ( 例では err.ldf ) にコメントとして記入されている必要があります。

- ◆ 一部の Sun ONE の objectClasses 定義にはネーミング属性がありません。これらの objectClasses を追加すると、eDirectory で次のエラーが発生することがあります。

LDAP エラー : 80 ( NDS エラー : ネーミングがあいまいです (-651) )

このエラーは、Sun ONE では eDirectory とは異なるネーミングルールの決定のメソッドが使用されていることが原因で発生します。

これを解決するには、次の 3 つのいずれかのオプションを使用できます。

### オプション 1:

エラーの原因となっている各 objectClasses を確認し、それぞれに有効なネーミング属性を追加します。

次に例を示します。

ネーミング属性 [cn] をオブジェクトクラス「netscapeMachineData」に追加するには、X-NDS\_NAMING フラグが包含されるように err.ldf ファイル内のエントリ ( 次の例で強調表示されている部分 ) を次のように変更します。

```
dn: cn=schema
changetype: modify
add: objectClasses
objectClasses: (2.16.840.1.113730.3.2.32 NAME 'netscapeMachineData'
DESC 'iPlanet defined objectclass' SUP top STRUCTURAL MAY 'cn'
X-NDS_NAMING 'cn' )-
```

### オプション 2:

エラーの原因となっている各 objectClasses を確認し、該当するものをすべて AUXILIARY または ABSTRACT に変更します。

次に例を示します。

オブジェクトクラス「netscapeMachineData」のオブジェクトクラス定義を「STRUCTURAL」から「AUXILIARY」に変更するには、次のように err.ldf ファイルのエントリ ( 次の例で強調表示されている部分 ) を変更します。

```
dn: cn=schema
changetype: modify
```



```
add: objectClasses

objectClasses: ( 2.16.840.1.113730.3.2.32 NAME 'netscapeMachineData'
DESC 'iPlanet defined objectclass' SUP top AUXILIARY )

-
```

オブジェクトクラス「netscapeMachineData」のオブジェクトクラス定義を「STRUCTURAL」から「ABSTRACT」に変更するには、次のように `err.ldf` ファイルのエントリ ( 次の例で強調表示されている部分 ) を変更します。

```
dn: cn=schema

changetype: modify

add: objectClasses

objectClasses: ( 2.16.840.1.113730.3.2.32 NAME 'netscapeMachineData'
DESC 'iPlanet defined objectclass' SUP top ABSTRACT )

-
```

### オプション 3:

eDirectory の Top の定義に `cn` を追加すると、すべての `objectClasses` の潜在的なネーミング属性になります。

`cn` を Top に追加するには 2 つの方法があります。

#### ◆ 方法 1:

次のようなファイルを作成し、`topsch.ldf` と命名します。

```
version: 1

dn:cn=schema

changetype :modify

delete:objectclasses

objectclasses : ( 2.5.6.0 NAME 'top' STRUCTURAL )

-

add:objectclasses

objectclasses : (2.5.6.0 NAME 'top' STRUCTURAL MAY cn)
```


次の Novell インポート / エクスポート変換コマンドラインを使用します。

```
ice -SLDIF -f LDIF_file_name -DLdap -s eDirectory_server -p
eDirectory_port -d eDirectory_Admin_DN -w eDirectory_password
```

次に例を示します。

```
ice -SLDIF -f topsch.ldf -DLdap -s edir_srv2 -p edir_port2 -d
cn=admin,o=org -w pwd1
```

#### ◆ 方法 2:

1. Novell iManager で、[役割およびタスク] ボタン  をクリックします。
2. [スキーマ] > [属性の追加] の順にクリックします。
3. [使用可能なクラス] リストで [トップ] を選択してから、[OK] をクリックします。

4. [使用可能なオプション属性] リストで、[CN] をダブルクリックします。

5. [OK] をクリックします。

- ◆ 一部の objectClass 定義には、必須属性リストとして *userPassword* が含まれる場合があります。これらの objectClasses を eDirectory に追加すると、次のエラーが発生します。

LDAP エラー : 16 ( 該当する属性はありません )

このエラーを解決するには、objectClass 定義を変更して、ndsLoginProperties から新しい objectClass を継承し、必須属性リストから userPassword 属性を削除します。

次に例を示します。

必須属性リストに userPassword が含まれる objectClass の場合 :

```
version : 1
dn:cn=schemaz
changetype: modify
add: objectClasses
objectClasses: ( 0.9.2342.19200300.100.4.19 NAME 'simpleSecurityObject'
DESC '
Standard LDAP objectClass' SUP top STRUCTURAL MUST userPassword )
```

次のように変更する必要があります ( 最終行の変更に注意してください )。

```
version : 1
dn: cn=schema
changetype: modify
add: bjectClasses
objectClasses: ( 0.9.2342.19200300.100.4.19 NAME 'simpleSecurityObject'
DESC '
Standard LDAP objectClass' SUP (ndsLoginProperties $ top) STRUCTURAL )
```

### 手順 3: LDIF ファイルのインポート

次の Novell インポート / エクスポート変換コマンドを使用して変更されたスキーマ比較 LDIF ファイル ( 例では err.ldf ) をインポートします。

```
ice -e error_file -SLDIF -f modified_LDIF_file -DLdap -s eDirectory_server -p eDirectory_port -d eDirectory_Admin_DN -w eDirectory_password
```

次に例を示します。

```
ice -e errors.ldf -SLDIF -f err.ldf -DLdap -s edir_srv2 -p edir_port2 -d cn=admin,o=org -w pwd1
```

## ICE を使用した、アクティブディレクトリスキーマの Novell eDirectory への移行

ICE を使用して、スキーマをアクティブディレクトリから Novell eDirectory に移行すると、あいまいなネーミングエラー (-651) というエラーメッセージが表示され、スキーマの「Computer」オブジェクトクラスへの移行が失敗します。

これを解決するには、次の手順を実行します。

[547 ページの「手順 1: スキーマキャッシュの更新操作を実行する」](#)

[548 ページの「手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する」](#)

[550 ページの「手順 3: LDIF ファイルのインポート」](#)

## 手順 1: スキーマキャッシュの更新操作を実行する

ICE を使用して、スキーマをアクティブディレクトリから Novell eDirectory へ移行し、次のように ICE のエラーログオプション (-e) が提供されていることを確認します。

```
ice -e error_file -S ldap -s Active_Directory_server -p Active_Directory_port
-d Active_Directory_full_admin_context -w Active_Directory_password -D ldap
-s eDirectory_server -p eDirectory_port -d eDirectory_full_admin_context -w
eDirectory_password
```

次に例を示します。

```
ice -e err.ldf -S ldap -s activesrv1 -p activeport1 -d cn=admin,o=company -w
activepwd -D ldap -s edirsrv2 -p edirport2 -d cn=admin,o=company -w edirpwd
```

## 手順 2: エラーを解決するためにエラー LDIF ファイルを訂正する

失敗したエントリは次のように err.ldf ファイルに記述されています。

```
dn: cn=schema

changetype: modify

delete: objectclasses

objectclasses: ( 2.16.840.1.113719.1.1.6.1.4 NAME 'computer' )
-

add: objectclasses

objectclasses: ( 2.16.840.1.113719.1.1.6.1.4 NAME 'computer' SUP (device $
user ) STRUCTURAL MAY (operator $ server $ status $ cn $ networkAddress $
local PolicyFlags $ defaultLocalPolicyObject $ machineRole $ location $
netbootInitialization $ netbootGUID $ netbootMachineFilePath $ siteGUID $
operatingSystem $ operatingSystemVersion $ operatingSystemServicePack $
operatingSystemHotfix $ volumeCount $ physicalLocationObject $ dnshostName $
policyReplicationFlags $ managedBy $ rIDSetReferences $ catalogs $
netbootSIFFile $ netboot MirrorDataFile ) X-NDS_NOT_CONTAINER '1' X
-NDS_NONREMOVABLE '1' X-NDS_NAME 'Computer' )
-

-
```

次の例で強調表示されている部分のように、エラーファイル (例では err.ldf) でこのエントリを編集して、「Computer」オブジェクトクラスの定義にあるスーパーリアオブジェクトクラスのリストから「user」オブジェクトクラスを削除します。

```
dn: cn=schema

changetype: modify

delete: objectclasses

objectclasses: ( 2.16.840.1.113719.1.1.6.1.4 NAME 'computer' )
-

add: objectclasses

objectclasses: ( 2.16.840.1.113719.1.1.6.1.4 NAME 'computer' SUP device
STRUCTURAL MAY (operator $ server $ status $ cn $ networkAddress $ local
PolicyFlags $ defaultLocalPolicyObject $ machineRole $ location $
netbootInitialization $ netbootGUID $ netbootMachineFilePath $ siteGUID $
operatingSystem $ operatingSystemVersion $ operatingSystemServicePack $
```

```
operatingSystemHotfix $ volumeCount $ physicalLocationObject $ dnsHostName  
$ policyReplicationFlags $ managedBy $ rIDSetReferences $ catalogs $  
netbootSIFFile $ netbootMirrorDataFile ) X-NDS_NOT_CONTAINER '1' X  
-NDS_NONREMOVABLE '1' X-NDS_NAME 'Computer' )
```

-

### 手順 3:LDIF ファイルのインポート

次の ICE コマンドを使用して、変更されたエントリをインポートします。

```
ice -S ldif -f LDIF_file -D ldap -s Novell_eDirectory_server -p port_number  
-d full_admin_context -w password
```

次に例を示します。

```
ice -S ldif -f err.ldf -D ldap -s edirsrv1 -p edirport1 -d cn=admin,o=company  
-w pwd1
```

# A

## NMAS の注意事項

この付録では、次のトピックについて説明します。

- ◆ 553 ページの「独立したパーティションとしてのセキュリティコンテナの設定」
- ◆ 553 ページの「複数のセキュリティコンテナを持つツリーのマージ」

### 独立したパーティションとしてのセキュリティコンテナの設定

NMAS™(Novell® Modular Authentication Services) は、Novell eDirectory™ ツリー全体に適用されるポリシーに依存しています。eDirectory ツリーは、実際にはセキュリティドメインとして機能します。セキュリティポリシーは、ツリー内のすべてのサーバで使用できる必要があります。

NMAS には、NetWare® 5.1 以降の eDirectory ツリーの [Root] から作成された、セキュリティコンテナ内の認証ポリシーとログインメソッドの設定データが格納されています。この情報は、NMAS を使用可能なすべてのサーバで読み込みアクセスできる必要があります。セキュリティコンテナの目的は、ログイン、認証、キー管理などのセキュリティプロパティに関するグローバルポリシーを保持することです。

NMAS により、独立したパーティションとしてセキュリティコンテナを作成し、作成したコンテナを広い範囲で複製することを推奨します。このパーティションは、ツリー内の信頼性の高い複数のサーバでのみ、読み書き可能なパーティションとして複製することをお勧めします。

**注：**セキュリティコンテナはグローバルポリシーを格納しており、サーバでは eDirectory ツリーに指定したセキュリティポリシー全般が変更される可能性があるため、書き込み可能なレプリカを配置するサーバを選択する場合には注意が必要です。NMAS を使用してユーザがログインするには、ユーザオブジェクトのレプリカが NMAS サーバ上に存在する必要があります。

### 複数のセキュリティコンテナを持つツリーのマージ

一方のツリーまたは両方のツリーにセキュリティコンテナがインストールされている eDirectory ツリーをマージする場合、特に注意が必要です。この手順は時間を要する複雑な作業になる可能性があるため、本当に実行する必要があるかどうか確認してください。

**重要：**この説明では、Novell Certificate Server™ 2.21 以前、Novell Single Sign-on 2.x、および NMAS 2.x のツリーを対象としています。

複数のセキュリティコンテナを持つツリーをマージするには次を実行します。

- 1 iManager で、マージするツリーを指定します。
- 2 ソースツリーにするツリーと、ターゲットツリーにするツリーを指定します。  
ソースツリーおよびターゲットツリーについて、次のセキュリティ上の配慮事項に注意してください。
  - ◆ ソースツリーの組織の認証局によって署名された証明書は削除すること。

- ◆ ソースツリーの組織の認証局は削除すること。
- ◆ ソースツリー上のNovell SecretStore®に保存されたユーザのシークレットは削除すること。
- ◆ ソースツリー内のすべての NMAS ログインメソッドを削除し、ターゲットツリーに再インストールすること。
- ◆ ツリーをマージする際に、ソースツリー内に存在したすべての NMAS ユーザを再登録すること。
- ◆ ソースツリー内に存在したすべてのユーザおよびサーバに、ツリーをマージする際に新しい証明書を作成すること。
- ◆ ソースツリー内に存在したすべてのユーザのシークレットを SecretStore に再インストールすること。

ソースツリーおよびターゲットツリーの両方が Security というコンテナをツリーのルート直下に保持していない場合、またはツリーのうち一方だけが Security コンテナを保持している場合、これ以上の処置は不要です。その他の場合には、このセクションの残りの手順を続けます。

## ツリーのマージ前に実行する製品固有の操作

このセクションでは、次の情報について説明します。

- ◆ [554 ページの「Novell Certificate Server」](#)
- ◆ [555 ページの「Novell Single Sign-On」](#)
- ◆ [556 ページの「NMAS」](#)
- ◆ [556 ページの「Novell セキュリティドメインインフラストラクチャ」](#)
- ◆ [557 ページの「その他のセキュリティ固有の操作」](#)

### Novell Certificate Server

Novell Certificate Server(PKIS(Public Key Infrastructure Services)の後継バージョン)がソースツリー内のサーバのいずれかにインストールされている場合、次の手順を実行する必要があります。

**注：**製品の使用状況によっては、指定されたオブジェクトと項目が存在しない可能性があります。次の手順で指定されたオブジェクトや項目がソースツリーに存在しない場合には、手順を省略できます。

- 1** ソースツリー内のルート認証局証明書は、ターゲットツリーにインストールする必要があります。

ルート認証局証明書は、ルート認証局コンテナに含まれる、ルート認証局オブジェクトに格納されています。ルート認証局コンテナはツリー内の任意の場所に作成できます。ただし、セキュリティコンテナ内のルート認証局コンテナに存在するルート認証局証明書は、ソースツリーからターゲットツリーに手動で移動する必要があります。

- 2** ターゲットツリーにルート認証局証明書をインストールします。

**2a** ソースツリーのセキュリティコンテナでルート認証局を選択します。

**2b** ソースツリーで使用されている正確な名前(手順2a)で、ルート認証局コンテナをターゲットツリーのセキュリティコンテナに作成します。

**2c** ソースツリーで、選択したルート認証局コンテナのルート認証局オブジェクトを開き、証明書をエクスポートします。

**重要：**次の手順で使用するため、選択した場所とファイル名を記憶しておきます。

- 2d** ターゲットツリーで、手順 **2b** で作成したコンテナのルート認証局オブジェクトを作成します。ソースツリーと同じ名前を指定し、証明書を求められたら、手順 **2c** で作成したファイルを指定します。
- 2e** ソースツリーのルート認証局オブジェクトを削除します。
- 2f** 選択したルート認証局コンテナ内のすべてのルート認証局オブジェクトがターゲットツリーにインストールされるまで、手順 **2c** から手順 **2e** を繰り返します。
- 2g** ソースツリーのルート認証局コンテナを削除します。
- 2h** すべてのルート認証局コンテナがソースツリー内で削除されるまで、手順 **2a** から手順 **2f** を繰り返します。
- 3** ソースツリーの組織の認証局を削除します。
- 組織の認証局オブジェクトは、セキュリティコンテナ内に存在します。
- 重要:** ソースツリーの組織の認証局によって署名された証明書は、この手順以降は使用できません。これには、ソースツリーの組織の認証局によって署名された、サーバ証明書とユーザ証明書も含まれます。
- 4** ソースツリーの組織の認証局によって署名された証明書を持つ、ソースツリー内の暗号化キーオブジェクト (KMO) をすべて削除します。
- 他の組織の認証局によって署名された証明書を持つ、ソースツリー内の暗号化キーオブジェクトは引き続き有効で、削除する必要はありません。
- 暗号化オブジェクトに署名している CA の識別情報がわからない場合は、暗号化オブジェクトのプロパティページにある [証明書] タブの [ルート認証局証明書] セクションを参照します。
- 5** ソースツリーの組織の認証局によって署名された、ソースツリー内のユーザ証明書をすべて削除します。
- ソースツリー内のユーザがすでに証明書とプライベートキーをエクスポートしている場合、エクスポートされた各証明書とキーは引き続き使用できます。手順 **3** を実行した後では、eDirectory 内に残っているプライベートキーと証明書は使用できません。
- 証明書を持つユーザごとに、ユーザオブジェクトのプロパティを開きます。[セキュリティ] タブの [証明書] セクションの下に、そのユーザのすべての証明書を示すテーブルが表示されます。発行者として組織の認証局が設定されている証明書はすべて削除してください。
- ソースツリーの組織の認証局のホストとなっているサーバに、Novell Certificate Server 2.0 以降がインストールされている場合のみ、ユーザ証明書はソースツリーに存在します。

## Novell Single Sign-On

Novell Single Sign-on がソースツリー内のサーバのいずれかにインストールされている場合、ソースツリーのユーザ用 Novell Single Sign-on シークレットをすべて削除する必要があります。

ソースツリーで Novell Single Sign-on を使用するすべてのユーザについて、ユーザオブジェクトのプロパティを開きます。[セキュリティ] タブの [SecretStore] セクションの下に、そのユーザのすべてのシークレットが表示されます。表示されたシークレットをすべて削除します。

**注:** 製品の使用状況によっては、指定されたオブジェクトと項目が存在しない可能性があります。指定されたオブジェクトや項目が存在しない場合には、この手順を省略できます。

NMAS がソースツリー内のサーバにインストールされている場合、次の手順を実行する必要があります。

**注：**製品の使用状況によっては、指定されたオブジェクトと項目が存在しない可能性があります。指定されたオブジェクトや項目が存在しない場合には、手順を省略できます。

- 1** ターゲットツリーに、ソースツリーに存在してターゲットツリーには存在しない、NMAS ログインメソッドをインストールします。

必要なすべてのクライアントとサーバのログインコンポーネントをターゲットツリーに適切にインストールするには、Novell のオリジナルソースやベンダー提供のソースを使用して、すべて新しいログインメソッドをインストールすることを推奨します。

メソッドは既存のサーバファイルから再インストールできますが、通常、Novell のパッケージやベンダー提供のパッケージを使用してクリーンインストールを行う方が簡単で確実です。

- 2** ソースツリーで以前に確立されたログインシーケンスをターゲットツリーで使用できるようにするには、対象のログインシーケンスを移行します。

**2a** ConsoleOne で、ソースツリーのセキュリティコンテナを選択します。

**2b** ログインポリシーオブジェクトを右クリックし、[プロパティ] をクリックします。

**2c** [定義済みログインシーケンス] ドロップダウンリストに表示されたログインシーケンスごとに、使用するログインメソッド (右側のウィンドウに表示) をメモします。

**2d** ターゲットツリー内のセキュリティコンテナを選択し、手順 **2c** でメモしたログインメソッドと同じものを使用して、ログインシーケンスを複製します。

**2e** 手順が終了したら、[OK] をクリックします。

- 3** ソースツリーの NMAS ログインセキュリティ属性を削除します。

**3a** ソースツリーのセキュリティコンテナで、ログインポリシーオブジェクトを削除します。

**3b** ソースツリーの許可されたログインメソッドコンテナで、ログインポリシーオブジェクトを削除します。

**3c** ソースツリーの許可されたログインメソッドコンテナを削除します。

**3d** ソースツリーの許可されたポストログインメソッドコンテナで、ログインポリシーオブジェクトを削除します。

**3e** ソースツリーの許可されたポストログインメソッドコンテナを削除します。

## Novell セキュリティドメインインフラストラクチャ

Novell Certificate Server 2.x 以降、Novell Single Sign-on、NMAS、NetWare 5.1 以降、または eDirectory 8.5 以降がソースツリー内のサーバのいずれかにインストールされている場合、Novell セキュリティドメインインフラストラクチャ (SDI) がインストールされます。SDI がインストールされている場合、次の手順を実行します。

**注：**製品の使用状況によっては、指定されたオブジェクトと項目が存在しない可能性があります。指定されたオブジェクトや項目が存在しない場合には、手順を省略できます。



- 1 W0 オブジェクトとソースツリーの KAP コンテナを削除します。  
KAP コンテナは、セキュリティコンテナ内に存在します。W0 オブジェクトは、KAP コンテナ内に存在します。
- 2 ソースツリー内のすべてのサーバで、`sys:\system\nici\nicisdi.key` ファイルを削除し、セキュリティドメインインフラストラクチャ (SDI) キーを削除します。  
**重要:** ソースツリー内のすべてのサーバで、このファイルを削除したことを確認します。

## その他のセキュリティ固有の操作

ソースツリー内にセキュリティコンテナが残っている場合、ツリーをマージする前に、セキュリティコンテナを削除します。

## ツリーのマージを実行する

`ndsmerge` ユーティリティを使用して、eDirectory ツリーをマージします。詳細については、[8 章 197 ページの「Novell eDirectory ツリーのマージ」](#) および [559 ページの付録 B、「Novell eDirectory の UNIX コマンドとその使用法」](#) を参照してください。

## ツリーのマージ後に実行する製品固有の操作

このセクションでは、次の情報について説明します。

- ◆ [557 ページの「Novell セキュリティドメインインフラストラクチャ」](#)
- ◆ [558 ページの「Novell Certificate Server」](#)
- ◆ [558 ページの「Novell Single Sign-On」](#)
- ◆ [558 ページの「NMAS」](#)

## Novell セキュリティドメインインフラストラクチャ

W0 オブジェクトがマージ前にターゲットツリーに存在していた場合、ターゲットツリー内に存在していたサーバが使用していたセキュリティドメインインフラストラクチャ (SDI) キーを、ソースツリー内に存在していたサーバにインストールする必要があります。

この手順を実行する一番簡単な方法は、SDI キー (`sys:\system\nici\nicisdi.key` ファイル) を保持していたソースツリー内のすべてのサーバに、Novell Certificate Server 2.52 以降をインストールすることです。Novell Certificate Server がすでにサーバにインストールされている場合でも、この手順を実行する必要があります。

マージ前に、ターゲットツリーに W0 オブジェクトが存在せず、ソースツリー内には存在していた場合は、マージ後のツリーに SDI を再インストールする必要があります。

この手順を実行する一番簡単な方法は、マージ後のツリー内のサーバに Novell Certificate Server 2.52 以降をインストールすることです。Novell Certificate Server は、SDI キー (`sys:\system\nici\nicisdi.key` ファイル) を保持していたサーバにインストールする必要があります。また、作成されたツリー内の他のサーバにもインストールできます。

Novell Certificate Server のインストールの詳細については、『[Novell Certificate Server Administration Guide \(Novell Certificate Server 管理ガイド\)](#) (<http://www.novell.com/documentation/japanese/crt252/index.html>)』を参照してください。

## Novell Certificate Server

Novell Certificate Server を使用している場合、ツリーのマージ後に、ソースツリー内に存在していたサーバとユーザに必要な応じて証明書を再発行します。

ユーザオブジェクトを含むパーティションのレプリカを保持していたすべてのサーバに、Novell Certificate Server 2.52 以降をインストールすることを推奨します。

サーバに証明書を発行するには、Novell Certificate Server 2.52 がインストールされている必要があります。

Novell Certificate Server 2.52 以降が、組織の認証局のホストとなっているサーバにインストールされている必要があります。詳細については、『*Novell Certificate Server Administration Guide (Novell Certificate Server 管理ガイド)* (<http://www.novell.com/documentation/japanese/crt252/index.html>)』を参照してください。

## Novell Single Sign-On

Novell Single Sign-on を使用している場合、ツリーのマージ後に、ソースツリー内に存在していたユーザの SecretStore シークレットを必要に応じて再作成する必要があります。

## NMAS

NMASを使用している場合、ツリーのマージ後に、ソースツリー内に存在していたNMASユーザを必要に応じて再登録する必要があります。

詳細については、『*Novell Modular Authentication Service Administration Guide (Novell Modular Authentication Service 管理ガイド)* (<http://www.novell.com/documentation/japanese/nmas22/index.html>)』を参照してください。

# B

## Novell eDirectory の UNIX コマンドとその使用方法

この章では、Novell® eDirectory™ 8.7.3 の Linux、Solaris、AIX、および HP-UX 用コマンドおよびその使用方法を表示します。

- ◆ [559 ページの「一般コマンド」](#)
- ◆ [561 ページの「LDAP 固有のコマンド」](#)

### 一般コマンド

このセクションでは、UNIX での eDirectory コマンドとその使用方法を表示します。

注: コマンドの使用法の詳細については、コマンドのマニュアルページを参照してください。

コマンド	説明	使用法
ndsbackup	eDirectory オブジェクトのアーカイブを作成し、eDirectory オブジェクトを追加または抽出	<pre>ndsbackup c [fevwXR] [ndsbackupfile] [exclude_file] [Replica_servername] [-a admin_user] [-I include_file]...[eDirectoryobject]  ndsbackup r [fevwXR] [ndsbackupfile] [exclude_file] [Replica_servername] [-a admin_user] [-I include_file]...[eDirectoryobject]  ndsbackup t [fevXR] [ndsbackupfile] [exclude_file] [Replica_servername] [-a admin_user] [-I include_file]...[eDirectoryobject]  ndsbackup x [fevwXR] [ndsbackupfile] [exclude_file] [Replica_servername] [-a admin_user] [-I include_file]...[eDirectoryobject]  ndsbackup s [evXR] [exclude_file] [Replica_servername] [-a admin_user] [-I include_file]...[eDirectoryobject]</pre>

コマンド	説明	使用法
ndsconfig	Novell eDirectory の環境設定	<pre>ndsconfig new [-m module_name] [-i] [-S server_name] [-t tree_name] [-n context] [-d path_for_dib] [-L ldap_port] [-l ssl_port] [-o http_port] [-O https_port] [-e] -a admin_name</pre> <pre>ndsconfig def [-m module_name] [-i] [-S server_name] [-t tree_name] [-n context] [-d path_for_dib] [-L ldap_port] [-l ssl_port] [-o http_port] [-O https_port] [-e] -a admin_name</pre> <pre>ndsconfig add [-m module_name] [-S server_name] [-t tree_name] [-p IP_address] [-n context] [-d path_for_dib] [-L ldap_port] [-l ssl_port] [-o http_port] [-O https_port] [-e] -a admin_name</pre> <pre>ndsconfig rm [-a admin_name]</pre> <pre>ndsconfig upgrade [-a admin_name]</pre> <pre>ndsconfig {set valuelist   get [paramlist]   get help [paramlist]}</pre>
ndsd	NDS <sup>®</sup> デーモン	<pre>/usr/sbin/ndsd [-f config_file]</pre> <p><b>注:</b> Solaris を再起動する前に ndsd を停止する必要があります。「/etc/init.d/ndsd stop」と入力します。</p>
ndsmonitor	HTTP を使用して、Novell eDirectory ツリー内のサーバの監視および診断を実行	<pre>/usr/bin/ndsmonitor [-l   -u]</pre>
ndslogin	Novell eDirectory の認証を確認するための診断ユーティリティ	<pre>ndslogin [-t treename] [-h hostname[:port]] [-p password] [-s] userFDN</pre>
ndsmerge	2 つの Novell eDirectory ツリーをマージするユーティリティ	<pre>ndsmerge [-m target_tree target_admin source_admin [target_container]] [-c] [-t] [-r target_tree source_admin]</pre>
ndsrepair	レコード、スキーマ、バインダリオブジェクト、および外部参照など、Novell eDirectory データベースの問題を修復および修正するためのユーティリティ	<pre>ndsrepair { -U   -E   -C   -P [-Ad]   -S [-Ad]   -N   -T   -J entry_id   --version} [-F filename] [-A yes_no] [-O yes no]</pre> <pre>ndsrepair -R [-l yes no] [-u yes no] [-m yes no] [-i yes no] [-f yes no] [-d yes no] [-t yes no] [-o yes no] [-r yes no] [-v yes no] [-c yes no] [-F filename] [-A yes no] [-O yes no]</pre>
ndssch	Novell eDirectory スキーマ拡張ユーティリティ	<pre>ndssch [-h hostname[:port]][-t treename] adminFDN schemafile ...</pre> <pre>ndssch [-h hostname[:port]][-t treename] [-d adminFDN schemafile [schema_description] ...</pre>
ndssnmp	Novell eDirectory の SNMP サービスモジュール	<pre>/usr/bin/ndssnmp</pre>

コマンド	説明	使用方法
ndssnmpconfig	SNMP トラップ環境設定ユーティリティ	<code>ndssnmpconfig -h [hostname[:port]] -p password -a userFDN -c command</code>
ndssnmpsa	eDirectory SNMP サブエージェントデーモン	<code>/usr/bin/ndssnmpsa</code>
ndsstat	サーバ情報を表示するユーティリティ	<code>ndsstat [-h hostname[:port]] {-r -s}</code>
ndstrace	サーバのデバッグメッセージを表示するユーティリティ	<code>/usr/bin/ndstrace</code> <code>ndstrace [-l -u -c 'command1;.....']</code>
nds-uninstall	Novell eDirectory をアンインストールするユーティリティ	<code>nds-uninstall [-c component1 [-c component2]...] [-h]</code>
nldap	LDAP Services for eDirectory デーモン	<code>/usr/sbin/nldap</code>
nmasinst	NMAS™ 環境設定ユーティリティ	<code>nmasinst -i admin.context treename</code> <code>nmasinst -addmethod admin.context_treename config.txt_path</code>
npki	Novellパブリックキーインフラストラクチャサービス	<code>/usr/sbin/npki</code>

## LDAP 固有のコマンド

コマンド	説明	使用方法
ldapconfig	LDAP サーバおよびLDAP グループオブジェクトを設定するユーティリティ	<code>ldapconfig get [...]   set attribute-value-list [-t treename   -p hostname[:port]] [-w password] [-a user FDN] [-f]</code>  <code>ldapconfig [-t treename   -p hostname[:port]] [-w password] [-a user FDN] [-V] [-R] [-H] [-f] -v attribute,attribute2...</code>  <code>ldapconfig [-t treename   -p hostname[:port]] [-w password] [-a admin FDN] [-V] [-R] [-H] [-f] -s attribute=value,...</code>
ldapadd	LDAP サーバからエントリを追加または変更	<code>ldapmodify [-a] [-c] [-C] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldap-port] [-P version] [-Z[Z]] [-f file]</code>  <code>ldapadd [-c] [-C] [-l] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldapport] [-P version] [-Z[Z]] [-f file]</code>
ldapdelete	LDAP サーバからエントリを削除	<code>ldapdelete [-n] [-v] [-c] [-r] [-l] [-C] [-M] [-d debuglevel] [-e key filename] [-f file] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldapport] [-Z[Z]] [dn]...</code>

コマンド	説明	使用法
ldapmodify	LDAP サーバからエントリを追加または変更	<pre> ldapmodify [-a] [-c] [-C] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldap-port] [-P version] [-Z[Z]] [-f file]  ldapadd [-c] [-C] [-l] [-M] [-P] [-r] [-n] [-v] [-F] [-l limit] [-M[M]] [-d debuglevel] [-e key filename] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldapport] [-P version] [-Z[Z]] [-f file] </pre>
ldapmodrdn	LDAP のエントリの相対識別名 (RDN) 変更ツール	<pre> ldapmodrdn [-r] [-n] [-v] [-c] [-C] [-l] [-M] [-s newsuperior] [-d debuglevel] [-e key filename] [-D binddn] [[-W]   [-w passwd]] [-h ldaphost] [-p ldapport] [-Z[Z]] [-f file] [dn newrdn] </pre>
ldapsearch	LDAP の検索ツール	<pre> ldapsearch [-n] [-u] [-v] [-t] [-A] [-T] [-C] [-V] [-M] [-P] [-L] [-d debuglevel] [-e key filename] [-f file] [-D binddn] [[-W]   [-w bindpasswd]] [-h ldaphost] [-p ldapport] [-b searchbase] [-s scope] [-a deref] [-l time limit] [-z size limit] [-Z[Z]] filter [attrs....] </pre>
ndsindex	Novell eDirectory データベースインデックスの作成、リスト表示、一時停止、再開、または削除に使用するユーティリティ	<pre> ndsindex list [-h &lt;hostname&gt;] [-p &lt;port&gt;] -D &lt;bind DN&gt; -W   [-w &lt;password&gt;] [-l limit] -s &lt;eDirectory Server DN&gt; [-Z[Z]] [&lt;indexName1&gt;, &lt;indexName2&gt;.....]  ndsindex add [-h &lt;hostname&gt;] [-p &lt;port&gt;] -D &lt;bind DN&gt; -W   [-w &lt;password&gt;] [-l limit] -s &lt;eDirectory Server DN&gt; [-Z[Z]] &lt;indexDefinintion1&gt; [&lt;indexDefinintion2&gt;.....]  ndsindex delete [-h &lt;hostname&gt;] [-p &lt;port&gt;] -D &lt;bind DN&gt; -W   [-w &lt;password&gt;] [-l limit] -s &lt;eDirectory Server DN&gt; [-Z[Z]] &lt;indexName1&gt; [&lt;indexName2&gt;.....]  ndsindex resume [-h &lt;hostname&gt;] [-p &lt;port&gt;] -D &lt;bind DN&gt; -W   [-w &lt;password&gt;] [-l limit] -s &lt;eDirectory Server DN&gt; [-Z[Z]] &lt;indexName1&gt; [&lt;indexName2&gt;.....]  ndsindex suspend [-h &lt;hostname&gt;] [-p &lt;port&gt;] -D &lt;bind DN&gt; -W   [-w &lt;password&gt;] [-l limit] -s &lt;eDirectory Server DN&gt; [-Z[Z]] &lt;indexName1&gt; [&lt;indexName2&gt;.....] </pre>

# C

## OpenSLP for eDirectory の設定

この付録では、ネットワーク管理者に向けて、Novell Client™ を使用しない OpenSLP for Novell® eDirectory™ インストールの適切な設定についての情報を提供します。

- ◆ 563 ページの「Service Location Protocol」
- ◆ 563 ページの「SLP の基本」
- ◆ 566 ページの「環境設定パラメータ」

### Service Location Protocol

OpenSLP は、IETF Service Location Protocol バージョン 2.0 標準のオープンソースの実装です。IETF Service Location Protocol バージョン 2.0 標準については、[IETF Request-For-Comments \(RFC\) 2608](http://www.ietf.org/rfc/rfc2608.txt?number=2608) (<http://www.ietf.org/rfc/rfc2608.txt?number=2608>) を参照してください。

OpenSLP ソースコードが提供するインタフェースでは、SLP v2 プロトコルの実装のほかに、プログラムで SLP 機能にアクセスする別の IETF 標準の実装があります。詳細は、[RFC 2614](http://www.ietf.org/rfc/rfc2614.txt?number=2614) (<http://www.ietf.org/rfc/rfc2614.txt?number=2614>) を参照してください。

SLP の動作の詳細を理解するためには、この 2 つのドキュメントを参照し、熟読してください。読みやすい文書ではありませんが、インターネットでの SLP の正しい設定を行うためには重要なドキュメントです。

OpenSLP プロジェクトの詳細については、[OpenSLP](http://www.OpenSLP.org) (<http://www.OpenSLP.org>) Web site and the [SourceForge](http://sourceforge.net/projects/openslp) (<http://sourceforge.net/projects/openslp>) の Web サイトを参照してください。OpenSLP の Web サイトには、環境設定に関する貴重なヒントを含んださまざまな文書があります。ただし、このガイドの作成時点では、これらのドキュメントの多くは未完成です。

### SLP の基本

Service Location Protocol では、次の 3 種類のコンポーネントが定義されています。

- ◆ ユーザエージェント (UA)
- ◆ サービスエージェント (SA)
- ◆ ディレクトリエージェント (DA)

ユーザエージェントは、クライアントがサービスを問い合わせたり、サービスがそれ自体をアドバタイズするためのプログラムインタフェースを提供します。ユーザエージェントはディレクトリエージェントに接続し、指定したスコープ内の指定したサービスクラスに登録されたサービスを問い合わせます。

サービスエージェントは、SLP で登録されたローカルサービスを持続的に格納し、維持する場所を提供します。サービスエージェントは主として、登録済みのローカルサービスをメモリ内データベースとして維持します。この場合、サービスはローカル SA がな

い限り SLP で登録できません。クライアントがサービスを検出するのは UA ライブラリ内のみですが、登録するには SA が必要です。これは主に、ディレクトリエージェントをリッスンして登録を維持するためには、登録済みサービスの存在を SA が定期的に表明する必要があるためです。

ディレクトリエージェントは、アドバタイズされたサービスに対して長期間持続的にキャッシュを提供し、ユーザエージェントがサービスを検索するためのアクセスポイントとなります。キャッシュ機能を提供する DA は、SA が新しいサービスをアドバタイズするのをリッスンし、これらの通知をキャッシュします。DA のキャッシュは短時間で完了します。ディレクトリエージェントは、期限切れのアルゴリズムを使用してエントリキャッシュを有効期限切れにします。ディレクトリエージェントが起動すると、持続的な格納領域 ( 通常はハードドライブ ) からキャッシュを読み込み、アルゴリズムに従ってエントリを有効期限切れにします。新しい DA が起動したり、キャッシュが削除されると、DA はこの条件を検出してリッスン中のすべての SA に特別な通知を送信します。SA は、DA が直ちにキャッシュを作成できるようにローカルデータベースをダンブします。

ディレクトリエージェントが存在しない場合、UA は SA が応答できる一般的なマルチキャスト方式のクエリを使用し、DA がキャッシュを作成するのとはほぼ同じ方法で、要求されたサービスのリストを作成します。このクエリによって返されるサービスのリストは、DA が提供するリストと比較すると不完全かつ局所的で、特に、多くのネットワーク管理者が使用するマルチキャスト方式でのフィルタ処理では、ブロードキャストおよびマルチキャストの対象がローカルサブネットのみに制限されるためです。

つまり、指定されたスコープに対してユーザエージェントが検索するものは、すべてディレクトリエージェントに依存します。

## Novell Service Location Providers

Novell のバージョンの SLP では、強力なサービスアドバタイズ環境を提供するため、SLP 標準が一部変更されます。しかし、このために一部の拡張性を犠牲にしています。

たとえば、サービスアドバタイズのフレームワークの拡張性を改善するために、サブネット上でのブロードキャストまたはマルチキャストのパケット数が制限されます。SLP の仕様では、これを管理するために、ディレクトリエージェントのクエリに関してサービスエージェントおよびユーザエージェントに制限を加えています。必要なスコープに対応するための最初に検出されたディレクトリエージェントは、サービスエージェント ( つまり結果的にローカルユーザエージェント ) がそのスコープ上の将来の要求すべてに使用するエージェントとなります。

Novell SLP を実装すると、クエリ情報の検索について既知のディレクトリエージェントをすべてスキャンします。スキャンの所要時間は 300 ミリ秒とかなり長く、したがって、約 3 ~ 5 秒以内で 10 台のサーバしかスキャンできません。SLP がネットワーク上で正しく設定されている場合にはこのような検索の必要はありません。OpenSLP では、ネットワークが実際に SLP トラフィック用に設定されていると見なされます。OpenSLP の応答タイムアウト値は Novell の SLP サービスプロバイダの応答タイムアウト値よりも大きい値です。ディレクトリエージェント数は、エージェントの情報が正確で完全であるかどうかに関係なく、最初に応答するディレクトリエージェントに制限されます。



## ユーザエージェント

ユーザエージェントの物理形式は、アプリケーションにリンクされたスタティックライブラリまたはダイナミックライブラリです。ユーザエージェントにより、アプリケーションは SLP サービスに対して問い合わせることができます。

ユーザエージェントは、アルゴリズムに従って、クエリの送信先になるディレクトリエージェントのアドレスを取得します。指定したスコープの DA アドレスを取得すると、ユーザエージェントはそのスコープから応答がなくなるまで同じアドレスを使用し続けます。応答がなくなると、ユーザエージェントはそのスコープに対する別の DA アドレスを取得します。ユーザエージェントは、指定されたスコープのディレクトリエージェントのアドレスを次の方法で検索します。

1. 現在の要求上のソケットハンドルが、指定したスコープの DA に接続されているかどうかをチェックする (要求がマルチパート要求の場合は、要求に対してキャッシュされた接続がすでに存在している可能性があります)。
2. 指定したスコープと一致している DA の、既知のローカル DA キャッシュをチェックする。
3. 指定したスコープでローカル SA に対して DA を確認する (その後キャッシュに新しいアドレスを追加します)。
4. 指定したスコープに一致する DA のネットワーク設定済みのアドレスを DHCP に問い合わせる (その後キャッシュに新しいアドレスを追加します)。
5. 既知のポートで DA の検出要求をマルチキャストする (その後キャッシュに新しいアドレスを追加します)。

スコープを指定しない場合、指定スコープは「デフォルト」になります。つまり、SLP 設定ファイルで静的に定義されたスコープがなく、クエリでスコープを指定していない場合は、使用されるスコープは「デフォルト」という単語になります。また、eDirectory の登録では eDirectory はスコープを指定しないことに注意してください。つまり、eDirectory で使用されるスコープは常に「デフォルト」というわけではありません。スコープが静的に設定されている場合、そのスコープがすべてのローカル UA 要求および SA 登録に対して、指定したスコープがない場合のデフォルトのスコープになります。

## サービスエージェント

サービスエージェントの物理形式は、ホストマシン上での個別のプロセスです。Win32 の場合は、slpd.exe がローカルマシン上のサービスとして実行されます。ユーザエージェントは、既知のポート上のループバックアドレスにメッセージを送信することによって、ローカルサービスエージェントを問い合わせます。

サービスエージェントは、潜在 DA アドレスに DA 検出要求を直接送信することにより、ディレクトリエージェントおよびそれがサポートするスコープリストを検出してキャッシュします。DA 検出要求は、次の方法で送信されます。

1. 静的に設定された DA アドレスをすべてチェックする (その後 SA の既知の DA キャッシュに新しい DA アドレスを追加します)。
2. DHCP から DA とスコープのリストを要求する (その後 SA の既知の DA キャッシュに新しいリストを追加します)。
3. 既知のポートで DA の検出要求をマルチキャストする (その後 SA の既知の DA キャッシュに新しいポートを追加します)。
4. DA によって定期的にブロードキャストされた DA のアドバタイズパケットを受信する (その後 SA の既知の DA キャッシュに新しいアドバタイズパケットを追加します)。

ユーザエージェントは常に最初にローカルサービスエージェントに対して問い合わせます。ローカルサービスエージェントの応答によってユーザエージェントが次の検出段階を続行するかどうかが決定的なため、このことは重要な点です (DHCP のこのケースについては、[565 ページの「ユーザエージェント」](#)の手順 3 および 4 を参照してください)。

## 環境設定パラメータ

`%systemroot%/slp.conf` ファイル内の各環境設定パラメータも、次のようにして DA の検出を制御します。

```
net.slp.useScopes = <comma delimited scope list>
net.slp.DAAddresses = <comma delimited address list>
net.slp.passiveDADetection = <"true" or "false">
net.slp.activeDADetection = <"true" or "false">
net.slp.DAActiveDiscoveryInterval = <0, 1, or a number of seconds>
```

**useScopes** オプションは、SA のアドバタイズ先のスコープ、および、サービスまたはクライアントアプリケーションで作成された登録またはクエリに指定したスコープが存在しない場合に、クエリが作成されるスコープを示します。**eDirectory** は常にデフォルトのスコープにアドバタイズし、問い合わせを行うため、このリストが **eDirectory** の登録およびクエリのデフォルトのスコープのリストになります。

**DAAddresses** はコンマで区切られた IP アドレスのリストで、アドレスは 10 進数とドットで表記されます。このアドレスが他のすべてに対して優先されます。設定された DA のこのリストが登録またはクエリのスコープをサポートしない場合、検出を無効にしない限りは、SA および UA はマルチキャスト方式で DA を検出します。

**passiveDADetection** オプションのデフォルトは「TRUE」です。ディレクトリエージェントは、設定に応じて定期的にそれ自身の存在をサブネットの既知のポート上にブロードキャストします。これらのパケットは **DAAdvert** パケットと名付けられます。このオプションに「FALSE」を設定した場合、ブロードキャスト方式のすべての **DAAdvert** パケットは SA に無視されます。

**activeDADetection** オプションのデフォルトも「TRUE」です。この設定により、SA はすべての DA に対して、指示された **DAAdvert** パケットで応答するように、定期的にブロードキャスト方式で要求できます。指示されたパケットはブロードキャストではありませんが、この要求に対する応答では SA に直接送信されます。このオプションに「FALSE」を設定した場合、SA は定期的な DA の検出要求をブロードキャストしません。

**DAActiveDiscoveryInterval** オプションは **try-state** パラメータです。デフォルト値は 1 です。これは、初期化の際に、SA が DA の検出要求を 1 回送る設定であることを意味する特別な値です。このオプションに 0 を設定すると、**activeDADetection** オプションに「FALSE」を設定した場合と結果は同じです。その他の値は、検出をブロードキャストする間隔を秒数で表します。

このオプションを正しく使用すると、サービスアドバタイズに使用するネットワーク帯域幅を適切に設定できます。ただし、デフォルト設定は平均的なネットワークで拡張性を最適化するように設計されています。

# D

## Novell eDirectory が DNS を使用する際の動作について

クライアントがサーバに Novell® eDirectory™ ツリーに存在しない完全修飾名 (admin.novell.novell\_inc など) の解決を要求する場合、またはツリー内での名前解決に Novell iManager for UNIX や eDirectory のインストールアプリケーションなどのスタンドアロンのアプリケーションを使用しており、サーバにはまだ接続していない場合、eDirectory はサービスディスカバリプロトコルを使用して名前を解決します。サービスディスカバリプロトコルはネットワークアプリケーションのクラスで、分散コンポーネントを使用してネットワーク内の必要なサービスを検索して使用できます。

eDirectory は、従来 SAP および SLP を使用してネットワークサービスを検索し、アドバタイズしてきました。DNS はディスカバリプロトコルとして eDirectory 8.7.1 に追加されました。この追加機能では、eDirectory が理解していないツリー名を問い合わせるために、コンピュータは eDirectory のディスカバリプロトコルを使用して、次の順序で検出を試みます。eDirectory がツリー名を理解しない理由は、接続しているサーバがツリーのコピーを保持していないことや、スタンドアロンのアプリケーションを使用していることによります。コンピュータは、スタンドアロンのアプリケーションを実行しているか、Novell iManager や ConsoleOne® などの JClient アプリケーションを実行しているか、またはサーバである場合があります。

1. ドメインネームシステム (DNS)
2. SLP (Service Location Protocol)
3. SAP (Service Advertising Protocol)

DNS プロトコルを使用すると、eDirectory は、サーバ名「prod\_server4.provo.novell.novell\_inc」など、渡されたままの名前を使用し、完全な名前の解決を試みます。eDirectory はディスカバリコンピュータの DNS 検索リストにそれぞれの名前を追加し、コンピュータの DNS サーバにその名前のアドレスが存在するかどうかを調べます。たとえば、ディスカバリコンピュータの DNS 検索リストに dev.novell.com および test.novell.com が含まれている場合、eDirectory は prod\_server4.provo.novell.novell\_inc.dev.novell.com および prod\_server4.provo.novell.novell\_inc.test.novell.com を検索します。

その後、eDirectory は渡された名前からコンポーネントを切り離します。たとえば、prod\_server4.provo.novell.novell\_inc を解決する場合、eDirectory は provo.novell.novell\_inc、novell.novell\_inc、novell\_inc の順に試みます。eDirectory は、各々の異なる検索コンテキストに対して検索を行い、最終的にツリーのルート of の単一コンポーネントに検索を試みます。クライアントは接続が成功するまで各アドレスを試行します。アドレスの試行には、DNS サーバから戻されたレコードの順序を使用します。検出を試みるコンピュータが eDirectory 8.7.1 以降を実行している限り、レプリカリング内のサーバがどのコードリビジョンを実行しても問題ありません。

eDirectory ツリー名は、クライアントが名前解決のために使用する DNS ドメインの下での A、AAAA、またはサービス (SRV) リソースレコードを使用する DNS に加えることをお勧めします。A または AAAA レコードを使用する場合、eDirectory サーバはデフォルトの 524 ポートで実行する必要があります。サーバが他のポートを使用する場合は、SRV レコードを使用します。

次のリソースレコードのサンプルでは、ツリー名は `novell_inc` で DNS 検索コンテキストは `provo.novell.com` です。

レコード	例
A	<code>novell_inc.provo.novell.com.IN A 192.168.1.2</code>
AAAA	<code>novell_inc.provo.novell.com.IN AAAA 4321:0:1:2:3:4:567:89ab</code>
SRV	<code>_ldap._tcp.novell_inc.provo.novell.com.SRV 0 0 389 server1.novell_inc.provo.novell.com SRV 10 0 389 server2.novell_inc.provo.novell.com</code>

冗長性を維持するため、または複数のホスト (レプリカリング内のサーバ) を A レコードに指定するためには、A レコードを 2 つ以上作成します。eDirectory は、それらすべてを検索します。A、AAAA、および SRV レコードの詳細については、[DNS レコードソース \(http://www.dns.net/dnsrd/r.html\)](http://www.dns.net/dnsrd/r.html) を参照してください。

対応するパーティションルートを保持しているものに対して、DNS サーバのレコードエントリを参照する必要はありません。ディスカバリコンピュータがツリーを認識しているサーバと通信でき次第、ツリー全体を参照して名前を解決できます。たとえば、DNS に `novell_inc` を設定する場合、`novell_inc` ルートを保持するサーバには `novell_inc` を設定する必要はありません。必要なのは `novell_inc` ツリー内の任意のサーバを参照することだけです。これは、ツリー内でそのサーバに接続できれば、そのサーバがツリー周辺でユーザを参照するためです。