

Novell exteNd Composer™ 5250 Connect

4.2

ユーザガイド

www.novell.com



Novell®

保証と著作権

Copyright ©1999, 2000, 2001, 2002, 2003 SilverStream Software, LLC. All rights reserved.

SilverStream ソフトウェア製品は、SilverStream Software LLC により著作権とすべての権利が保留されています。

SilverStream は SilverStream Software, LLC の登録商標です。Novell は、Novell, Inc. の登録商標です。

ソフトウェアとマニュアルの所有権、および特許、著作権、およびそれに関連するその他のすべての財産権は常に、単独で排他的に SilverStream とそのライセンサーに保留され、当該所有権と矛盾するいかなる行為も行わないものとします。本ソフトウェアは、著作権法と国際条約規定で保護されています。ソフトウェアならびにそのマニュアルからすべての著作権に関する通知とその他の所有権に関する通知を削除してはならず、ソフトウェアとそのマニュアルのすべてのコピーまたは抜粋に当該通知を複製しなければなりません。本ソフトウェアのいかなる所有権も取得するものではありません。

Jakarta-Regexp Copyright ©1999 The Apache Software Foundation. All rights reserved. Ant Copyright ©1999 The Apache Software Foundation. All rights reserved. Xalan Copyright ©1999 The Apache Software Foundation. All rights reserved. Xerces Copyright ©1999-2000 The Apache Software Foundation. All rights reserved. Jakarta-Regexp、Ant、Xalan、Crimson、および Xerces ソフトウェアは、The Apache Software Foundation によりライセンスを付与され、Jakarta-Regexp、Ant、Xalan、Crimson、および Xerces のソースおよびバイナリ形式での再配布および使用は、変更のあるなしにかかわらず、以下の条件が満たされることを前提として許可されます。1. ソースコードの再配布に上記の著作権に関する通知、条件のリスト、および以下の権利放棄に関する通知が記載されていること。2. バイナリ形式の再配布では上記の著作権に関する通知、条件のリスト、および以下の権利放棄に関する通知がマニュアルまたは配布の際に提供されるその他の資料、あるいはその両方に記載されていること。3. エンドユーザの資料には、適宜、以下の通知を再配布の際に含めてください。「この製品には、Apache Software Foundation (<http://www.apache.org>) により開発されたソフトウェアが含まれています」代わりに、この謝辞をソフトウェア自体に表示し、当該サードパーティに対する謝辞が通常表示される場所に表示することもできます。4. 「The Jakarta Project」、「Jakarta-Regexp」、「Xerces」、「Xalan」、「Ant」、および「Apache Software Foundation」は、書面による事前の許可なく、このソフトウェアから派生する製品を推薦したり、販売促進したりするのに使用してはなりません。書面による許可については、apache@apache.org <<mailto:apache@apache.org>> にお問い合わせください。5. 本ソフトウェアから派生する製品は「Apache」と呼ばれてはならず、「Apache」は The Apache Software Foundation の事前の書面による許可なくその名前に使用することはできません。本ソフトウェアは「現状のまま」提供され、いかなる明示的、暗黙の保証も行われるものではありません。販売可能性や特定の目的に対する適合性に対する暗黙の保証も行われません。いかなる場合でも、Apache Software Foundation またはその関係者はいかなる直接的、間接的、偶発的、特別な、免除的、または結果的な損害（代替品やサービスの調達、使用機会、データ、または利益の喪失、または業務の中断などを含む）についても、理論上責任がある場合でも、契約上の責任がある場合でも、厳密な責任、または瑕疵（怠慢などを含む）があった場合でも、ソフトウェアの使用の過程で生じ、当該損害の可能性を助言した場合であっても、責任をもちません。

Copyright ©2000 Brett McLaughlin & Jason Hunter. All rights reserved. ソースおよびバイナリ形式での再配布および使用は、変更のあるなしにかかわらず、以下の条件が満たされることを前提として許可されます。1. ソースコードの再配布に上記の著作権に関する通知、条件のリスト、および以下の権利放棄に関する通知が記載されていること。2. バイナリ形式の再配布では上記の著作権に関する通知、条件のリスト、および以下の権利放棄に関する通知がマニュアルまたは配布の際に提供されるその他の資料、あるいはその両方に記載されていること。3. 「JDOM」という名前は、書面による事前の許可なく、このソフトウェアから派生する製品を推薦したり、販売促進したりするのに使用してはなりません。書面による許可については、license@jdom.org <<mailto:license@jdom.org>> にお問い合わせください。4. 本ソフトウェアから派生する製品は「JDOM」と呼ばれてはならず、「JDOM」は JDOM Project Management (pm@jdom.org <<mailto:pm@jdom.org>>) の事前の書面による許可なくその名前に使用することはできません。本ソフトウェアは「現状のまま」提供され、いかなる明示的、暗黙の保証も行われるものではありません。販売可能性や特定の目的に対する適合性に対する暗黙の保証も行われません。いかなる場合でも、Apache Software Foundation またはその関係者はいかなる直接的、間接的、偶発的、特別な、免除的、または結果的な損害（代替品やサービスの調達、使用機会、データ、または利益の喪失、または業務の中断などを含む）についても、理論上責任がある場合でも、契約上の責任がある場合でも、厳密な責任、または瑕疵（怠慢などを含む）があった場合でも、ソフトウェアの使用の過程で生じ、当該損害の可能性を助言した場合であっても、責任をもちません。

Sun Microsystems, Inc. Sun, Sun Microsystems, Sun Logo Sun, Sun のロゴ, Sun Microsystems, JavaBeans, Enterprise JavaBeans, JavaServer Pages, Java Naming and Directory Interface, JDK, JDBC, Java, HotJava, HotJava Views, Visual Java, Solaris, NEO, Joe, Netra, NFS, ONC, ONC+, OpenWindows, PC-NFS, SNM, SunNet Manager, Solaris sunburst design, Solstice, SunCore, SolarNet, SunWeb, Sun Workstation, The Network Is The Computer, ToolTalk, Ultra, Ultracomputing, Ultraserver, Where The Network Is Going, SunWorkShop, XView, Java WorkShop, Java Coffee Cup のロゴ, Visual Java, および NetBeans は、米国およびその他の国の Sun Microsystems, Inc. の商標ならびに登録商標です。

Copyright ©2001 Extreme! Lab, Indiana University License. <http://www.extreme.indiana.edu>. 同社により許可が無料で、Indiana University ソフトウェアと関連する Indiana University のドキュメントファイル (「IU Software」) のコピーを取得したすべての人に、制限なく IU Software を取り扱うために付与されます。その際に、IU Software の使用、コピー、変更、マージ、公開、配布、サブライセンス、または販売、あるいはそれらのすべてに関する権利に制限はなく、IU Software が指定した人に以下の条件に基づき権利を付与します。上記の著作権に関する通知とその許可に関する通知は、IU Software のすべてのコピーおよび主要部分に含まれる必要があります。本 IU ソフトウェアは「現状のまま」提供され、いかなる明示的、暗黙の保証も行われるものではありません。販売可能性、特定の目的に対する適合性や権利侵害がないことに対する暗黙の保証も行われません。いかなる場合でも、作成者または著作権所有者は、契約上の責任がある場合でも、厳密な責任、または瑕疵 (怠慢などを含む) があつた場合でも、IU Software に関連して、または UI Software の使用やその他の取引の過程で生じた場合であっても、クレーム、損害、その他の責任について責任を持ちません。

本ソフトウェアは、著作権をもつ SSLava™ Toolkit の一部です。Copyright ©1996-1998 by Phaos Technology Corporation. All Rights Reserved.

Copyright © 1994-2002 W3C® (Massachusetts Institute of Technology, Institut National de Recherche Informatique et en Automatique, Keio University), all Rights Reserved. <http://www.w3.org/consortium/legal>. この W3C の成果物 (ソフトウェア、ドキュメント、またはその他の関連品目を含む) は、以下のライセンスの下で著作権所有者により提供されています。この成果物の取得、使用、またはコピー、あるいはそれらのすべてにより、ライセンシーは以下の条件を読み、理解し、遵守することに合意するものとします。本ソフトウェアとそのドキュメントの使用、コピー、変更、および配布は、変更のあるなしにかかわらず、いかなる目的でも無料または本契約で許可された使用料をもって許可されます。ただし、変更箇所を含む本ソフトウェアとドキュメントのすべてまたはその一部に以下のとおり記述することを前提とします。1. この通知の全文は、再配布物または派生物のユーザが見やすい場所に掲示しなければなりません。2. すべての前もって存在する知的所有権の放棄、通知、または条件。存在しない場合は、以下の形式の短い通知 (ハイパーテキストが望ましい、テキストでも良い) を再配布または派生コードの本文内で使用しなければなりません。[Copyright © [Date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>] 3. W3C のファイルに変更または修正を加えた場合はその日付を含む通知。(コードが派生する場所への URI を示すことをお勧めします。) 本ソフトウェアは「現状のまま」提供され、いかなる明示的、暗黙の保証も行われるものではありません。販売可能性、特定の目的に対する適合性やサードパーティの特許、著作権、商標またはその他の権利を侵害しないことに対する暗黙の保証も行われません。著作権の所有者は本ソフトウェアまたはマニュアルの使用の結果生じる、直接的、間接的、特殊な、または結果的な損害に対していかなる責任も負いません。著作権所有者の名前および商標は、特別な書面による事前の承諾なしにソフトウェアに関する広告や広報に使用してはなりません。本ソフトウェアおよび関連する資料の著作権の所有権は常に、著作権所有者に帰属するものとします。

米国 Novell, Inc.
1800 South Novell Place
Provo, UT 85606

www.novell.com

5250 Connect ユーザガイド
2003 年 1 月
000-000000-000

オンラインマニュアル： この製品およびその他の Novell 製品のオンラインマニュアルや更新情報については、
www.novell.com/documentation を参照してください。

Novell の商標

Novell および jBroker は Novell, Inc. の登録商標です。また Novell exteNd は Novell, Inc. の商標です。

サードパーティ商標

Sun Microsystems, Inc. Sun, Sun Microsystems, Sun Logo Sun、Sun のロゴ、Sun Microsystems、JavaBeans、Enterprise JavaBeans、JavaServer Pages、Java Naming and Directory Interface、JDK、JDBC、Java、HotJava、HotJava Views、Visual Java、Solaris、NEO、Joe、Netra、NFS、ONC、ONC+、OpenWindows、PC-NFS、SNM、SunNet Manager、Solaris sunburst design、Solstice、SunCore、SolarNet、SunWeb、Sun Workstation、The Network Is The Computer、ToolTalk、Ultra、Ultracomputing、Ultraserver、Where The Network Is Going、SunWorkShop、XView、Java WorkShop、Java Coffee Cup のロゴ、Visual Java、および NetBeans は、米国およびその他の国の Sun Microsystems, Inc. の商標ならびに登録商標です。

目次

このガイドについて	9
1 exteNd Composer および 5250 User Interface へようこそ	11
はじめに	11
exteNd Composer Connect について	12
5250 (TDS) Connect とは	13
exteNd Composer の 5250 コンポーネントについて	14
5250 User Interface コンポーネントエディタを使用して作成できるアプリケーション	16
2 5250 コンポーネントエディタをお使いになる前に	17
サンプルトランザクション	17
5250 接続リソースの作成	18
システムメッセージの処理	23
スタイルシートリソースの作成	26
コンポーネントに対する XML テンプレートの作成	29
3 5250 コンポーネントの作成	31
5250 コンポーネントを作成する前に	31
5250 コンポーネントエディタウィンドウについて	34
5250 固有のメニューバー項目について	39
4 5250 アクションの実行	43
アクションについて	43
5250 専用アクションについて	44
5250 セッションの記録	47
5250 Multi Row ウィザードについて	59
アニメーションツールの使用	81
Component with Connection アクション	86
ネイティブ環境ペインでのスタイルシートの使用	91
5250 コンポーネントエディタでの他のアクションの使用	92
エラーおよびメッセージの処理	94
5 ログオンコンポーネント、接続、および接続プール	99
5250 端末セッションパフォーマンスについて	99
接続プールアーキテクチャ	100
5250 ログオン接続について	102
5250 ログオンコンポーネントについて	103
接続プールの作成	110
接続の作成	110
ログオンコンポーネントの作成	111
プール接続を使用したログオン接続の作成	113
セッション接続を使用したログオン接続の作成	120

5250 端末コンポーネントの作成	122
プールの管理	124
接続プール管理および配備済みのサービス	126
A Java コードページ	129
文字エンコーディングについて	129
B 5250 用語集	131
C テスト	133
アニメーションテストと配備テストの環境的相違	133
D 予約語	135

このガイドについて

目的

このガイドでは、5250 コンポーネントエディタと呼ばれる exteNd Composer 5250 Connect の使用方法について説明します。5250 コンポーネントエディタは、exteNd Composer で個別にインストールされるコンポーネントエディタです。

対象読者

このガイドの対象読者は、exteNd Composer を使用して、5250 アプリケーションを統合するサービスやコンポーネントを作成する開発者およびシステムインテグレータです。

前提条件

このガイドでは、読者が exteNd Composer の開発環境および配備オプションに精通しており、これらを使用したことがあるという前提で説明していきます。また、5250 環境を理解しておく必要もあります。

追加のドキュメント

Novell exteNd Composer の完全なマニュアルのセットは、Novell マニュアルの Web サイト (<http://www.novell.com/documentation-index/index.jsp>) を参照してください。

構成

このガイドは、次のように編成されています。

第 1 章「exteNd Composer および 5250 へようこそ」では、5250 コンポーネントエディタの定義および概要について説明します。

第 2 章「5250 コンポーネントエディタをお使いになる前に」では、5250 コンポーネントを作成するために必要な準備について説明します。

第3章「5250 コンポーネントの作成」では、コンポーネントエディタの異なる部分について説明します。

第4章「5250 アクションの実行」では、基本的な 5250 アクションの使用方法、および 5250 Multi Row Wizard について説明します。

第5章「ログオンコンポーネント、接続、および接続プール」では、ログオンコンポーネント、接続、および接続プールについて説明します。

付録 A 「Java コードページ」では、文字エンコーディング変換に関する情報を示します。

付録 B は、用語集です。

付録 C 「テスト」では、アニメーションテスト時と配備テスト時における環境的相違について説明します。

付録 D では、5250 でのみ使用される「予約語」について説明します。

このガイドで使用する表記規則

このガイドで使用する表記規則は、次のとおりです。

手順での「太字」のフォントは、次のアクション項目を示します。

- ◆ メニューの選択
- ◆ フォームの選択
- ◆ ダイアログボックス項目

太字の **Sans Serif** フォントは、次の項目に使用します。

- ◆ Uniform Resource Identifier
- ◆ ファイル名
- ◆ ディレクトリおよび部分的なパス名

「斜体」のフォントは、次の項目を示します。

- ◆ 入力する変数情報
- ◆ 新出の技術用語
- ◆ 他の Novell 出版物のタイトル

「モノスペース」のフォントは、次の項目を示します。

- ◆ メソッド名
- ◆ コードの例
- ◆ システム入力
- ◆ オペレーティングシステムオブジェクト

1

exteNd Composer および 5250 User Interface へようこそ

はじめに

『5250 Connect ガイド』へようこそ。このガイドは、Composer の全機能 (Connect コンポーネントエディタを除く) の使用方法が詳しく説明されている『exteNd Composer ユーザガイド』に付属しています。『ユーザガイド』をご覧になっていない場合は、このガイドを使用する前に読んで内容を確認してください。

exteNd Composer には、コネクタ (5250 など) ごとに異なるコンポーネントエディタが用意されています。各コンポーネントエディタ独自の機能は、それぞれのガイドで説明されています。

exteNd Composer を使用しており、コアコンポーネントエディタである XML Map コンポーネントエディタに精通しているユーザは、このガイドに従って 5250 コンポーネントエディタを簡単に使用することができます。

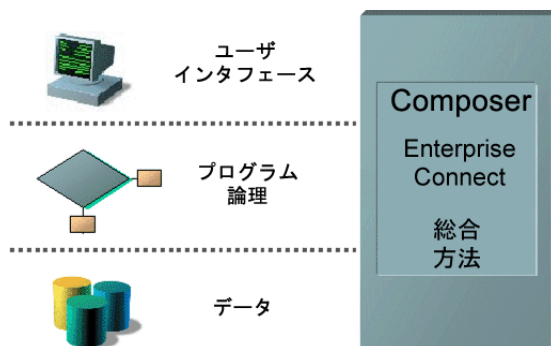
作業を始める前に、まず 5250 Connect を既存の exteNd Composer にインストールしておく必要があります。また、この Connect で作成されたサービスを Composer Enterprise Server 環境で実行するには、事前にこの Connect のサーバ側ソフトウェアを Composer Enterprise Server にインストールしておく必要があります。

注記： このコンポーネントエディタを正しく使用するには、IBM 5250 環境と、XML に対応させるアプリケーションに慣れ親しんでおく必要があります。

exteNd Composer Connect について

exteNd Composer は、単純なハブ & スポークアーキテクチャに基づいて構築されています。ハブは、XML ドキュメントを使用して要求を受け付け、XML に対応したアプリケーション上でこのようなドキュメントやインタフェースで変換プロセスを実行し、XML 応答ドキュメントを返す強力な XML 変換エンジンです。スポーク（つまり Connect）は、XML 対応でないデータのソースを「XML に対応させる」プラグインモジュールで、データをハブに送信して XML として処理します。これらのデータソースには、レガシーCOBOL/アプリケーションから、HTML ページに対するメッセージキューまで何でも使用できます。

exteNd Composer Connect は、情報ソースを XML に対応させるために各製品で使用されている統合方法に従って分類できます。統合方法は、インターネットベースのコンピュータアーキテクチャに対する現在のシステム設計において使用される主要な区分を反映したものです。exteNd Composer では、B2B のニーズとレガシーアプリケーションのアーキテクチャに応じて、ユーザインタフェースレベル、プログラム論理レベル、またはデータレベルでビジネスシステムを統合できます。

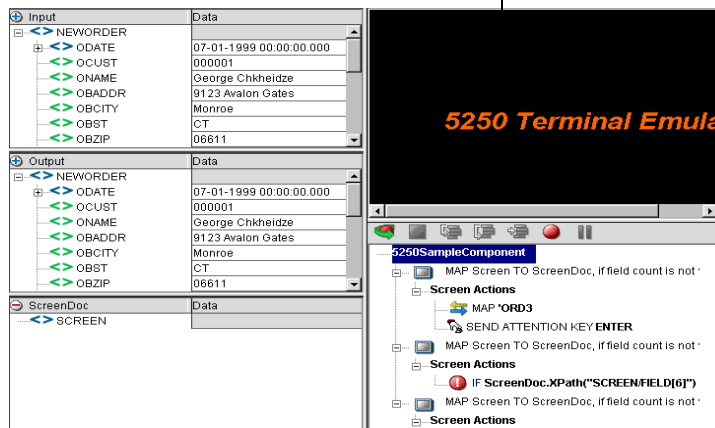


5250 (TDS) Connect とは

TDS (端末データストリーム) にフッキングすることによって User Interface 統合方法を使用する、5250 Connect XML に対応した IBM AS/400 レガシーデータです。5250 という用語は、IBM AS/400 中規模システムへの接続に使用される一般的な「ダム端末」タイプを指します。5250 TDS は、IBM AS/400 に接続する際、IBM の EBCDIC 文字エンコードスキームを使用します。1960 年代に開発された 5250 TDS は、当時の標準規格として採用され、現在にも受け継がれています。5250 TDS は、適切なアクションを実行するためにホストで実行されているアプリケーションによって解釈されるアテンションキー (例 : <Enter> キーやファンクションキー) を使用して、レガシーアプリケーションを操作できるようにします。ダム端末を通じて処理するという事は、すべてのデータが AS/400 コンピュータのデータを処理した結果であることを意味します。5250 端末エミュレーションソフトウェアによって、AS/400 との通信中にマイクロコンピュータまたは PC を 5250 タイプの端末のように動作させることができます。

5250 Connect を使用すると、IBM AS/400 で実行されているレガシーアプリケーションやそれらのビジネスロジックを、インターネットプロセス、エクストラネットプロセス、またはイントラネットプロセスで利用することができます。端末セッションでの場合と同様にアプリケーションを移動したり、キー入力する代わりに XML 要求ドキュメントを使用して問い合わせや更新を画面に送り込んだり、アプリケーション画面から返されたメッセージを使用して端末での場合と同様に同じ決断を下したり、リクエストに返すことのできる XML ドキュメントにデータや応答を移動するか処理を続行することができます。5250 画面は、5250 コンポーネントエディタのネイティブ環境画面に表示されます。

ネイティブ環境ペインに 5250 画面が表示される

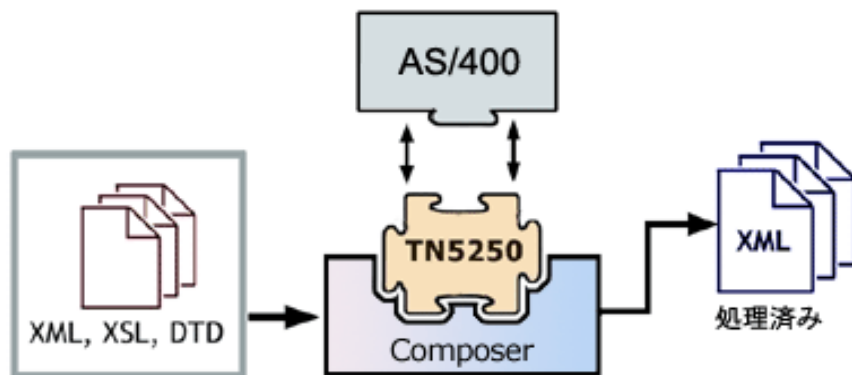


exteNd Composer の 5250 コンポーネントについて

XML Map コンポーネントと同様に、5250 コンポーネントは、2つの異なる XML テンプレート (つまり、要求 XML ドキュメントと応答 XML ドキュメント) 間でデータをマップ、変換、および転送するために設計されています。ただし、5250 コンポーネントは、AS/400 への TN5250 接続を確立し、DOM のデータを端末セッション用に処理し、端末セッションの結果を出力 DOM にマッピングする処理に特化されています。ユーザは、出力 DOM を、独自の統合アプリケーションで利用できるよう自由に処理することができます。つまり、レガシーシステム自体を変更せずに、レガシーシステムからデータをキャプチャしたり、レガシーシステムにデータをプッシュすることが可能です。

5250 コンポーネントでは、単純なデータ操作 (XML ドキュメントから AS/400 トランザクションへのデータのマップや転送など) を実行したり、5250 トランザクションの「スクリーンスクレーピング」を実行したりして、XML ドキュメントにデータを配置することができます。さらに、同じ行が繰り返されている画面や、複数のデータ画面をマッピングしたり操作したりするなど、高度な処理も実行できます。これらの処理は、exteNd Composer では複数行トランザクション、複数画面トランザクションと呼ばれます。5250 コンポーネントには XML Map コンポーネントの全機能が備わっており、XSL の処理、メールの送信、および HTTP プロトコルを使用した XML ドキュメントのポストと受信を実行できます。

次の図は、5250 コンポーネントが TN5250 接続によって AS/400 のデータと通信するしくみを示しています。



5250 User Interface コンポーネントエディタを使用して作成できるアプリケーション

EDI User Interface コンポーネントエディタを使用すると、exteNd Composer で、作成中の XML 統合アプリケーションに、5250 ベースの端末操作をサポートするあらゆるビジネスアプリケーションを含めることができます（詳細については、『exteNd Composer ユーザガイド』を参照してください）。たとえば、定期的に更新されるデータベースから製品の説明、画像、価格、および在庫情報を取得して Web ブラウザに表示するアプリケーションがあるとします。5250 コンポーネントエディタを使用すると、動作中のシステムから最新の製品情報を取得したり、データベースから静的情報（画像など）を取得し、個別の情報ソースの情報をマージしてから、ユーザに対して表示できます。これにより、内部ユーザと外部ユーザの両方に同じ最新情報が提供されます。

2

5250 コンポーネントエディタをお使いになる前に

サンプルトランザクション

このドキュメントでは、PART、GORD、および MENU という 3 種類のトランザクションのサンプルを例として使用します。これらのトランザクションは、オペレータが使用するトランザクションの典型です。PART トランザクションは、オペレータが SKU 番号を使用してデータベースへの問い合わせを行うシナリオです。GORD トランザクションは、1 つまたは複数のアイテムへの注文が発行されるシナリオです。MENU トランザクションは、オペレータがメニュー駆動型のアプリケーションを移動して特定の画面にアクセスするシナリオです。PART、GORD、MENU の各トランザクションは、実際のシナリオと同様の操作を行う Composer サービスを作成する方法を示すために使用します。

5250 コンポーネントの作成に使用される一般的な手順

5250 コンポーネントを作成する方法はいくつもありますが、単純な 5250 コンポーネントの作成に使用される一般的な手順は次のとおりです。

- 1 トランザクション用の XML テンプレートを作成します。
- 2 接続リソースを作成します。
- 3 コンポーネントを作成します。
- 4 記録モードを入力し、トランザクションへ移動します。
- 5 入力ドキュメントを画面にドラッグアンドドロップします。
- 6 キーボード操作でトランザクションを処理します。
- 7 画面の結果を出力ドキュメントにドラッグアンドドロップします。
- 8 記録を停止します。

5250 接続リソースの作成

5250 コンポーネントを作成する前に、AS/400 トランザクションにアクセスするための接続リソースを作成する必要があります。

接続リソースについて

5250 コンポーネント用の接続リソースを作成するには、TN5250 接続を選択しなければなりません。IBM AS/400 環境に接続する場合は、TN5250 接続を使用できます。

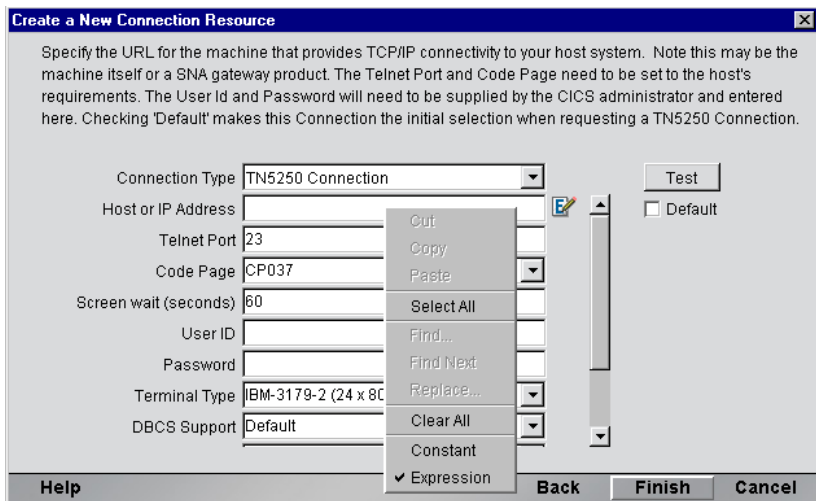
定数駆動型および式駆動型の接続について

接続パラメータの値は、定数または式としての方法のうちいずれかを使用して指定できます。定数ベースのパラメータでは、接続が使用されるたびに[Connection] ダイアログボックスに入力した値を使用します。式ベースのパラメータでは、ランタイム時に接続が使用されるたびに異なる値となり得る、プログラムの式を使用して値を設定できます。この場合、接続の動作は柔軟になり、接続の使用ごとにランタイム時の条件に対応できるようになります。

たとえば、TN5250 接続における式駆動型のパラメータの非常に単純な使用の 1 つは、ユーザ ID とパスワードを PROJECT 変数 (例: PROJECT.XPATH(“SERCONFIG/MyDeployUser”)) として定義することです。このようにすると、プロジェクトを配備する際に、Deployment Wizard で PROJECT 変数を最終配備環境に適切な値に更新できます。それとは正反対に、アプリケーションサーバで Java ビジネスオブジェクトを照会するカスタムスクリプトを使って、使用するユーザ ID とパスワードを決定することもできます。

➤ 定数駆動型から式駆動型にパラメータを切り替える

- 1 変更するパラメータフィールドでマウスを右クリックします。
- 2 コンテキストメニューから [Expression] を選択すると、エディタボタンが表示されるか、または有効になります。
- 3 ボタンをクリックし、ランタイム時に有効なパラメータ値を返す式を作成します。

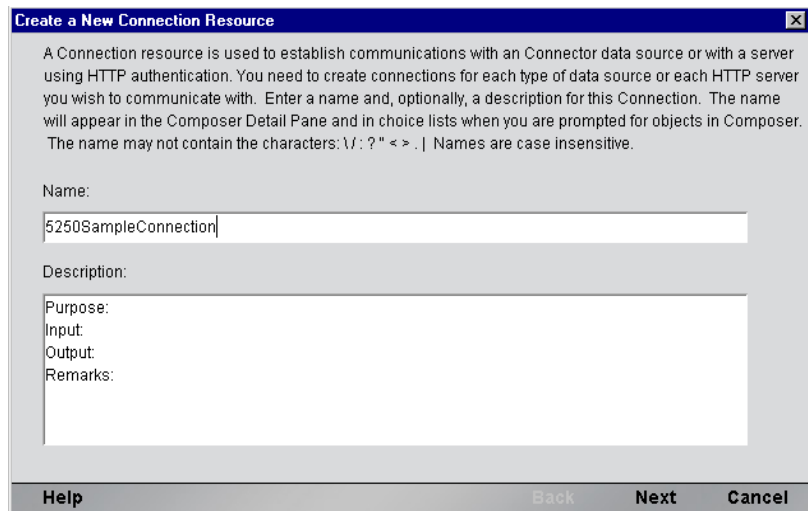


➤ 5250 接続リソースを作成する

- 1 Composer の **[File]** メニューから、**[New xObject]**、**[Resource]**、**[Connection]** の順に選択します。

注記： または、Composer ウィンドウのカテゴリペインで **[Connection]** を選択し、マウスを右クリックした後で **[New]** を選択することもできます。

[Create a New Connection Resource] ウィザードが表示されます。



- 2 **[Name]** に、接続オブジェクトの名前を入力します。

- 3 オプションとして、[Description] に説明のテキストを入力します。
- 4 [Next] をクリックします。[Create a New Connection Resource] ウィザードの 2 ページ目が表示されます。

Specify the URL for the machine that provides TCP/IP connectivity to your host system. Note this may be the machine itself or a SNA gateway product. The Telnet Port and Code Page need to be set to the host's requirements. The User Id and Password will need to be supplied by the CICS administrator and entered here. Checking 'Default' makes this Connection the initial selection when requesting a TN5250 Connection.

Connection Type: TN5250 Connection

Host or IP Address: []

Telnet Port: 23

Code Page: CP037

Screen wait (seconds): 60

User ID: []

Password: []

Terminal Type: IBM-3179-2 (24 x 80)

DBCS Support: Default

DBCS Code Page: CP950

Version 2.7 Compatibility:

Test

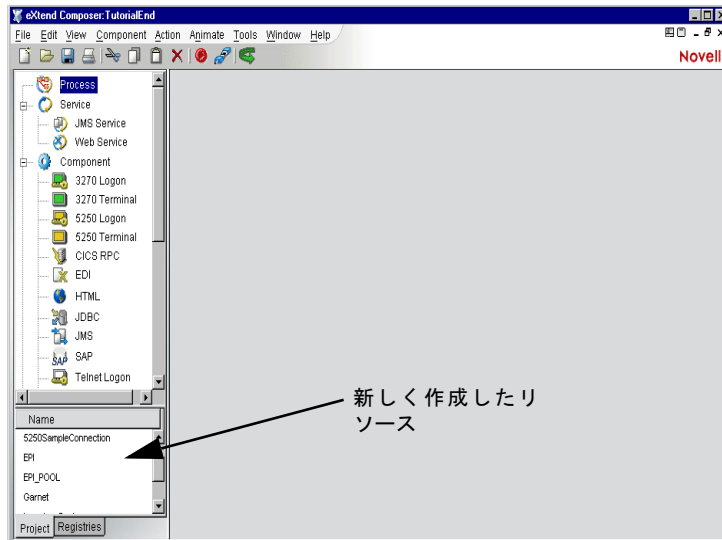
Default

Advanced...

Help Back Finish Cancel

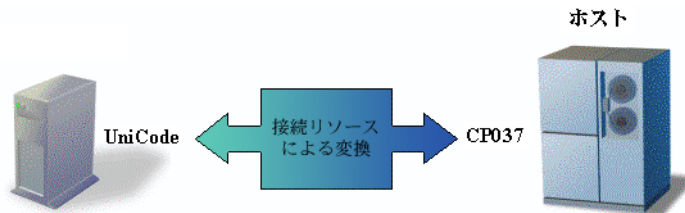
- 5 [Host or IP Address] フィールドに、接続先コンピュータの物理アドレスまたは別名を入力します。この情報はシステム管理者に問い合わせてください。これは、各ホストファイルに定義されています。
- 6 [Telnet Port] フィールドに、ポートの番号を入力します。デフォルトのポート番号は 23 です。
- 7 [Code Page] フィールドで、ドロップダウンリストからコードページを選択します (22 ページの「コードページサポートについて」を参照)。
- 8 [Screen wait (seconds)] フィールドには、5250 端末コンポーネントが [Map Screen Action] ペインで次の画面を待つ時間が秒単位で表示されます。
- 9 [UserID] にユーザ ID を、[Password] にパスワードを入力します。これらは、接続の確立中にホストに対して実際には送信されず、単にここで定義されます (パスワードは暗号化されます)。ユーザは、ECMAScript からユーザ ID とパスワードにアクセスし、それらを変数として画面にマップできるようになります。このため、パスワードが誰かに表示されることはありません。
- 10 [Terminal Type] フィールドには、異なる画面サイズ (たとえば、24x80 および 27x32) を含む、5250 コンポーネントでサポートされる端末のタイプが表示されます。ドロップダウンリストボックスから、使用する端末のタイプを選択します。
- 11 [DBCS Support] フィールドのドロップダウンリストボックスで、[Default]、[Double Encoding]、[SO/SI Using Ox1F] のうちいずれかを選択します。

- 12 [DBCS Code Page] フィールドのドロップダウンリストボックスから、コードページを選択します。
- 13 [Version 2.7 Compatability] チェックボックスをオンにします。
- 14 この 5250 接続を、後に続く 5250 コンポーネントのデフォルトの接続にする場合は、[Default] チェックボックスをオンにします。
- 15 画面ハンドラを作成するには [Advanced] ボタンをクリックします。詳細については、次のセクションの「システムメッセージの処理」を参照してください。
- 16 [Finish] をクリックします。新しく作成されたリソース接続オブジェクトが、Composer 接続リソースの詳細ペインに表示されます。



コードページサポートについて

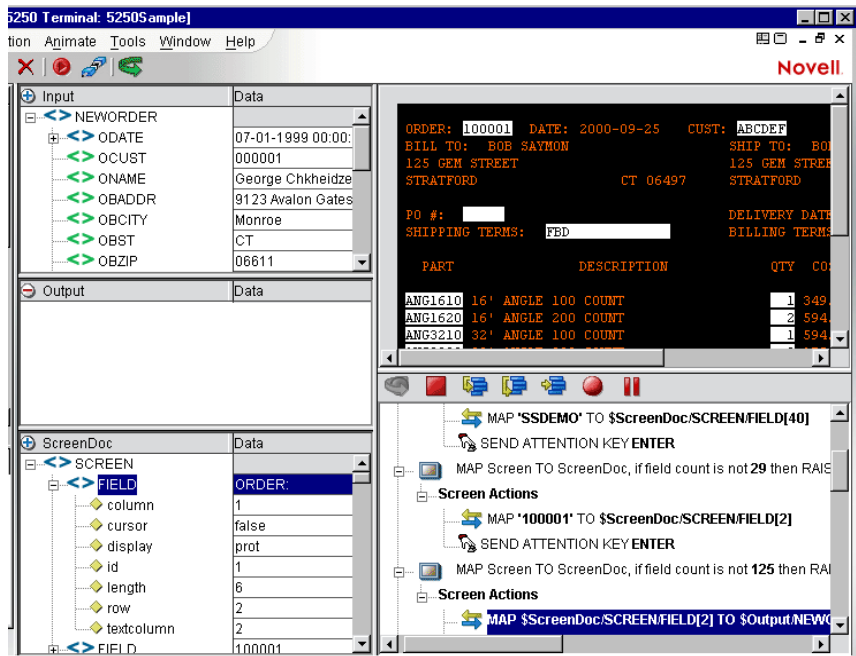
exteNd Composer 接続リソースのコードページサポートでは、exteNd Composer とその他のホストシステム間で送信された文字を変換する際に使用する文字エンコードスキームを指定できます。exteNd Composer データでは、Unicode 文字エンコード (Java および XML 標準) が使用されます。既存のレガシーホストシステムとその他のホストシステムでは、言語または使用に特定のさまざまな文字エンコードスキーム (つまり、コードページ) を使用します。これらのシステム間で互いに通信する場合は、それぞれの文字エンコードを変換するためのメカニズムが必要となります。これは、そのシステムにアクセスするのに使用される接続リソースのホストシステムで使用されるコードページを指定することによって、exteNd Composer で処理されます。文字エンコードの詳細については、付録 A、「Java コードページ」を参照してください。



システムメッセージの処理

5250 接続には、5250 コンポーネントの実行で問題となり得る、端末セッションで受信した予期しないメッセージ画面を処理するための特別な機能があります。このタイプのメッセージ画面の例としては、システム管理者が端末にシステムのシャットダウンまたはその他のイベントについて警告するための通知を送信した場合の画面があります。システムトランザクションと通信している 5250 コンポーネントの通常の実行では、この画面の表示は予期されず、また不要です。通常これらのシステムメッセージは、読んだ後にユーザが AID キーを押すと消えます。

TN5250 接続によって、特殊な画面ハンドラを定義し、コンポーネント処理の対象外にする画面に応答できます。単一の接続で複数の画面ハンドラを指定することもできます。接続リソースで画面ハンドラを指定することにより、ハンドラを一度指定すると、同じ接続を使用して複数のコンポーネントでそのハンドラを利用できるようになります。



注記： 接続の画面ハンドラはシステム管理者からのメッセージのみに限定されません。画面ハンドラを指定すると、端末のデータストリームで受信するなどの画面に対しても応答することができます。

➤ 画面ハンドラを作成するための一般的な手順

- 1 5250 接続リソースを使用する、作業用 5250 コンポーネントを作成します。

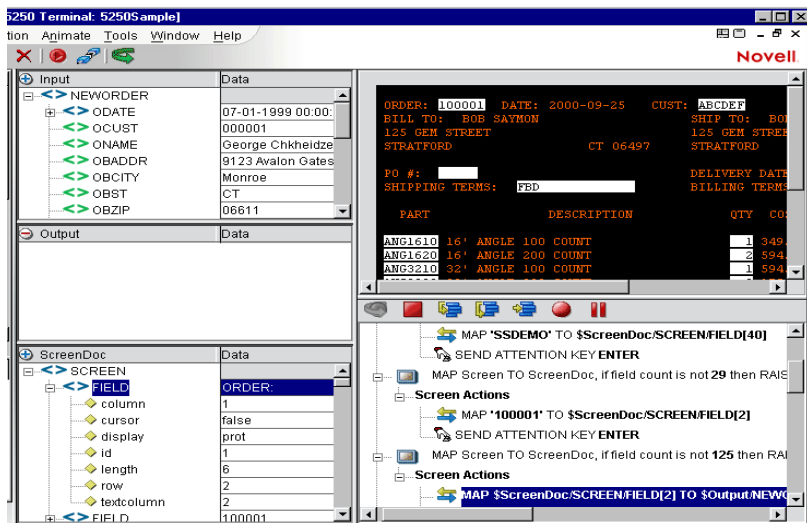
- 2 処理する画面を ScreenDoc DOM で再現し、キャプチャします。
- 3 画面でフィールドとデータの組み合わせを選択し、この画面を画面ハンドラとして一意に特定します。
- 4 画面を識別する式と画面を閉じるための AID キーを指定して、画面ハンドラを接続リソースに追加します。

➤ 画面ハンドラを作成するための詳細な手順

- 1 通常の方法で 5250 コンポーネントを作成します (つまり、接続とその他のリソースを必要に応じて作成し、次にコンポーネントを作成します)。
- 2 ScreenDoc DOM が表示されていない場合、メニューバーから **[View]**、**[Window Layout]** の順に選択し、次に **[XML Layout]** をクリックします。**[ScreenDoc]** を **[Visible]** リストに移動し、**[OK]** をクリックします。
- 3 コンポーネントを保存し、開いたままにします。次の手順に従って、処理する特定の画面をキャプチャします。

注記： この作業には、システムオペレータ / 管理者、または処理したいメッセージ画面の生成に携わる人の協力が必要です。

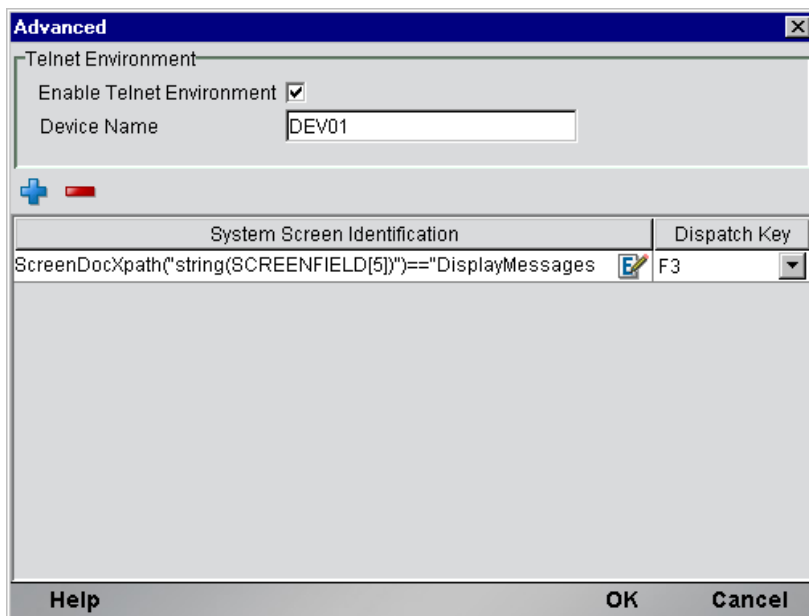
- a. Log in アクションを含む Map Screen の後にくる最初の Map Screen のアクションに、ブレイクポイントを設定します。(これがログオンコンポーネントである場合、KeepAlive または Logon セクションの最初のアクションにブレイクポイントを設定します)。
- b. <F5> キーを押して、コンポーネントをアニメーション表示します。
- c. <F9> キーを押して、ブレイクポイントを実行します。
- d. <F6> キーを押して、アニメーション表示を一時停止します。
- e. システムメッセージを開始します。
- f. <Enter> キーを押すと、ネイティブ環境ペインには、システムメッセージが表示され、Screen Doc ペインには、メッセージを XML で表したものが表示されます。



- 4 この画面を処理する画面として特定するためのデータを持つテストフィールドを選択します。たとえば、「Display Messages」というデータを持つ画面タイトルフィールドを選択します。
- 5 ScreenDoc で、テストフィールドの XML 表記にマウスのポインタを合わせ、XPath の場所の完全パス (例: ScreenDoc/SCREEN/FIELD[5]) を確認します。
- 6 <Shift>+<F5> キーを押してアニメーション表示を停止し、ホストシステムをサインオフします。次に、保存せずにコンポーネントを閉じます。
- 7 Composer ウィンドウで、コンポーネントの接続リソースを選択し、開きます。
- 8 [Connection Info] タブで [Advanced] ボタンをクリックして、[Advanced] ダイアログボックスを表示します。
- 9 [Enable Telnet Environment] チェックボックスをオンにして、Telnet プロトコルサポートを有効にします。
- 10 TN5250 Connect で使用されているデバイス名を [Device Name] に入力します。
- 11 [Add] ボタンをクリックし、画面ハンドラを指定する行を追加します。
- 12 式フィールドで、テストデータをテストフィールドの XPath と比較する式 (例: ScreenDoc.XPath("string(SCREEN/FIELD[5])")=" Display Messages") を入力します。

注記: 比較が適切に実行されるかどうかを確認するには、XPath 文字列関数を適用します。

[AID/DispatchKey] フィールドで、画面への応答として送信するキーを選択します。



- 13 [OK] をクリックしてハンドラを保存し、次に [OK] をクリックして接続を保存します。
- 14 Logon Map Screen アクションを通過したコンポーネントをアニメーション表示する、ブレークポイントを設定する、システムメッセージを開始する、アニメーションに Enter Send Key アクションを通過させる、などを操作を行い、システム画面ハンドラをテストします。

スタイルシートリソースの作成


5250 コネクタに関連付けられる追加のリソースとして、スタイルシートリソースがあります。これにより、ネイティブ環境ペインにエミュレーション画面を表示するためのスタイルシートを作成できます。

➤ スタイルシートリソースを作成する

- 1 Composer の [File] メニューから [New xObject] を選択し、次に [Resource]、[Terminal Style Sheet] の順に選択します。

注記： また、Composer のカテゴリペインの [Resource] セクションから、[Terminal Style Sheet] を選択し、次に、マウスの右ボタンをクリックしてから [New] を選択します。

Create a New Terminal Style Sheet ウィザードが表示されます。



Create a New Terminal StyleSheet Resource

In this Dialog you can edit the properties of the StyleSheet.

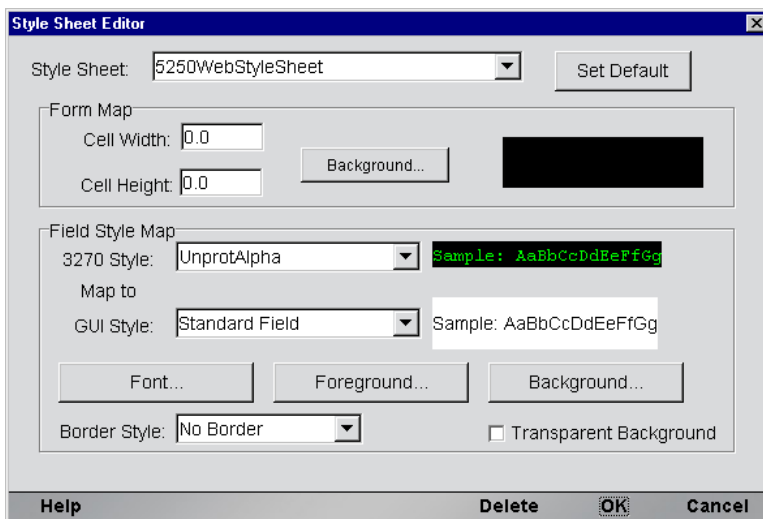
Name:
5250WebStyleSheet

Description:
Purpose:
Input:
Output:
Remarks:

Help Back Next Cancel

- 2 [Name] に、新しいスタイルシートの名前を入力します。オプションで、[Description] に説明テキストを入力することができます。

- 3 [Next] ボタンをクリックします。[Style Sheet Editor] ウィンドウが表示されます。



- 4 次に説明に従って、[Style Sheet Editor] ウィンドウでスタイルシートを設定します。

- ◆ **[Style Sheet]** - このドロップダウンリストから、ネイティブ環境ペインのエミュレーション画面を変更するスタイルシートを選択します。最初このフィールドには、Terminal Style Sheet ウィザードの最初のページで指定した名前が表示されます。新しいスタイルシートを作成するには、リストの名前の1つに新しい名前を上書きします。
- ◆ **[Set Default]** - このボタンを選択すると、現在選択されているスタイルシートがコンポーネントのデフォルトになります。
- ◆ **[Form Map]**
 - [Cell Width/Height]** - これらの設定で、フォントタイプの変更によって切り捨てられる可能性のある文字を修正します。
 - [Background]** - このボタンを選択して、スタイルシートの背景色のオプションを表示します。

◆ **[Field Style Map]**

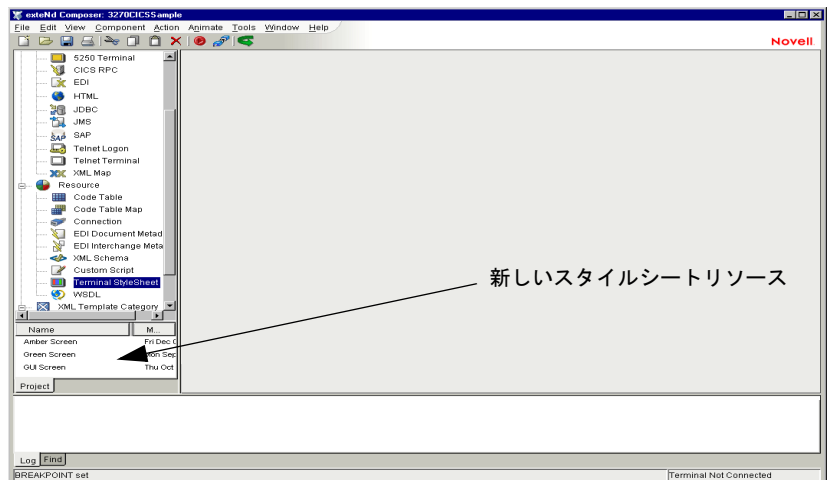
[5250 Style] - このコントロールでは、TDS で使用できるスタイルがリストで表示されます。これらの値を編集することはできません。新しく作成するスタイルにマップするスタイルを選択します。

[GUI Style] - このコントロールでは、作成した使用可能なスタイルが表示されます。新しいスタイルを作成するには、既存のスタイルに新しいスタイルを上書きし、該当するボタンを使用して、「フォント」、「前景」、および「バックグラウンド」を指定します。

[Border Style] - あらかじめ定義された 3 種類の境界のうち 1 つを、ドロップダウンリストから選択します。このコントロールを編集することはできません。

[Transparent Background] - GUI に透明のバックグラウンドを使用する場合は、このチェックボックスをオンにします。

- 5 **[OK]** をクリックします。Composer の詳細ペインに、新しく作成したスタイルシートリソースが表示されます。



コンポーネントに対する XML テンプレートの作成

接続リソースの他に、5250 コンポーネントでは、コンポーネントを設計するためのサンプルドキュメントを持つように、XML テンプレートをすでに作成していることが必要です。詳細については、『exteNd Composer ユーザガイド』の第 5 章「XML テンプレートの作成」を参照してください。

多くの場合、入力ドキュメントは、端末オペレータがトランザクションにインタラクティブに入力する可能性のあるデータを含むように設計されます。同様に、出力ドキュメントは、オペレータの入力の結果として画面に返されるデータを受信するよう設計されます。たとえば、ビジネスシナリオでは、端末オペレータは、アイテムの価格や購入が可能であるかどうかについて興味のある顧客から電話による問い合わせを受けます。顧客は、システムに入力する SKU 番号をオペレータに伝えます。これは、入力ドキュメントの SKU 番号に相当します。同様に、オペレータの端末が SKU に基づいたホストからの情報を受信すると、オペレータは顧客にこの情報を伝えます。これは出力ドキュメントに相当します。

また、コンポーネント設計によって別の xObject リソース (カスタムスクリプトやコードテーブルマップなど) が要求される場合は、5250 コンポーネントを作成する前にこれらのリソースを作成することが推奨されます。詳細については、『exteNd Composer ユーザガイド』を参照してください。

3

5250 コンポーネントの作成

5250 コンポーネントを作成する前に

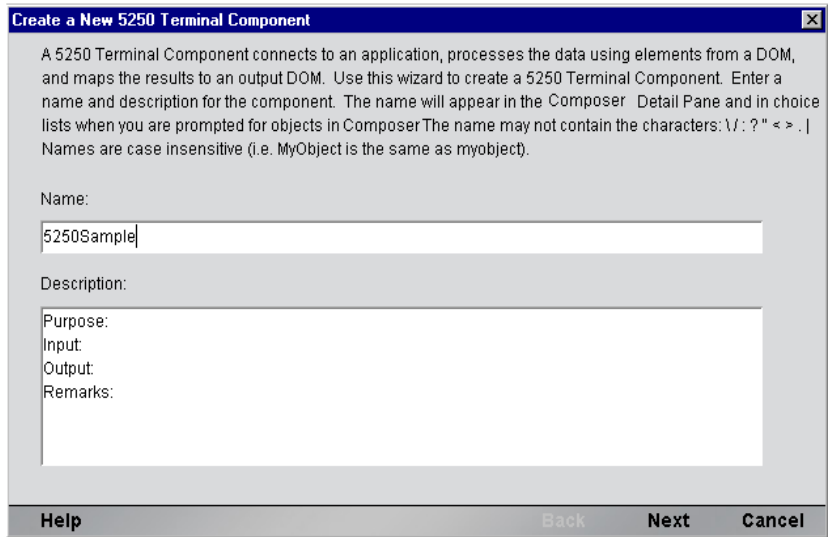
すべての exteNd Composer コンポーネントと同様に、5250 コンポーネントを作成する最初の手順は、必要な XML テンプレートを指定することです。詳細については、『Composer ユーザガイド』の「新しい XML テンプレートの作成」を参照してください。

XML テンプレートを指定すると、コンポーネントによって処理される入力および出力を表すテンプレートのサンプルドキュメントを使用して、コンポーネントを作成できます。

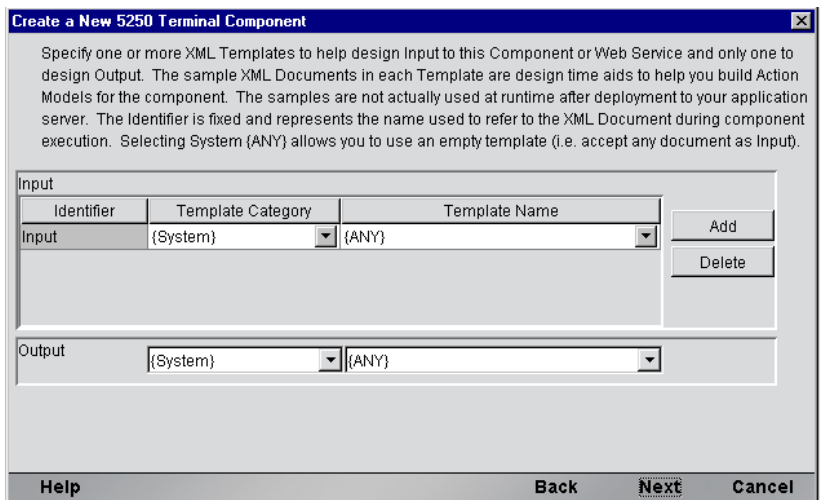
また、5250 コンポーネントの作成プロセスの一環として、5250 接続を選択する必要があります (または、新しく作成することもできます)。18 ページ「5250 接続リソースの作成」を参照してください。

➤ 新しい 5250 コンポーネントを作成する

- 1 [File] > [New xObject] > [Component] > [5250 Terminal] の順に選択します。Create a New 5250 Component ウィザードが表示されます。



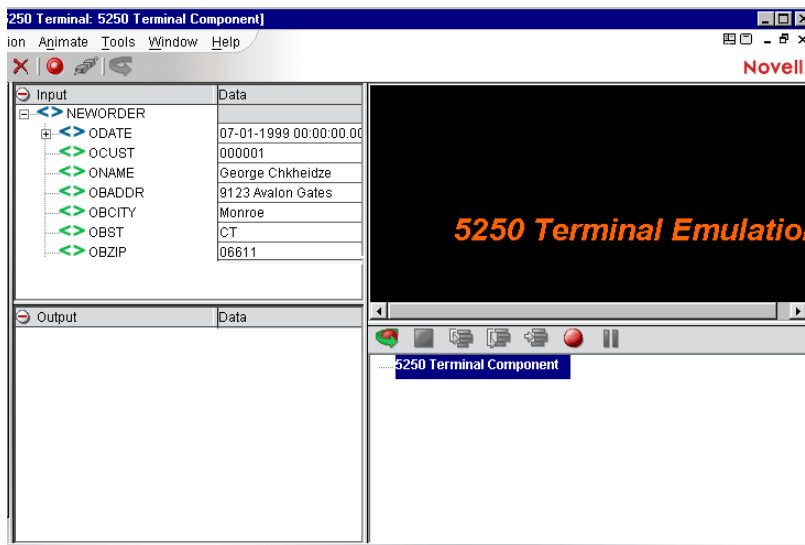
- 2 新しい 5250 コンポーネントの「名前」を入力します。
- 3 オプションとして、「説明」テキストを入力します。
- 4 [Next]をクリックします。Create a New 5250 Component ウィザードの XML プロパティ情報パネルが表示されます。



- 5 入力テンプレートおよび出力テンプレートを次のように指定します。
 - ◆ デフォルトのカテゴリと異なる場合は、「テンプレートカテゴリ」を選択します。

- ◆ 選択した「**テンプレートカテゴリ**」にある XML テンプレートのリストから「**テンプレート名**」を選択します。
 - ◆ 入力 XML テンプレートをさらに追加するには、[**Add**] をクリックして、それぞれに「**テンプレートカテゴリ**」と「**テンプレート名**」を選択します。
 - ◆ 入力 XML テンプレートを削除するには、エントリを選択して [**Delete**] をクリックします。
- 6** 出力としての XML テンプレートを選択します (出力 DOM の名前は「Output」です)。
- 注記:** 出力テンプレートとして {System}{ANY} を選択すると、構造が含まれない入力 XML テンプレートまたは出力 XML テンプレートを指定できます。詳細については、『Composer ユーザガイド』の「テンプレートを使用しない出力 DOM の作成」を参照してください。
- 7** [**Next**] をクリックします。Create a New 5250 Component ウィザードの接続情報パネルが表示されます。

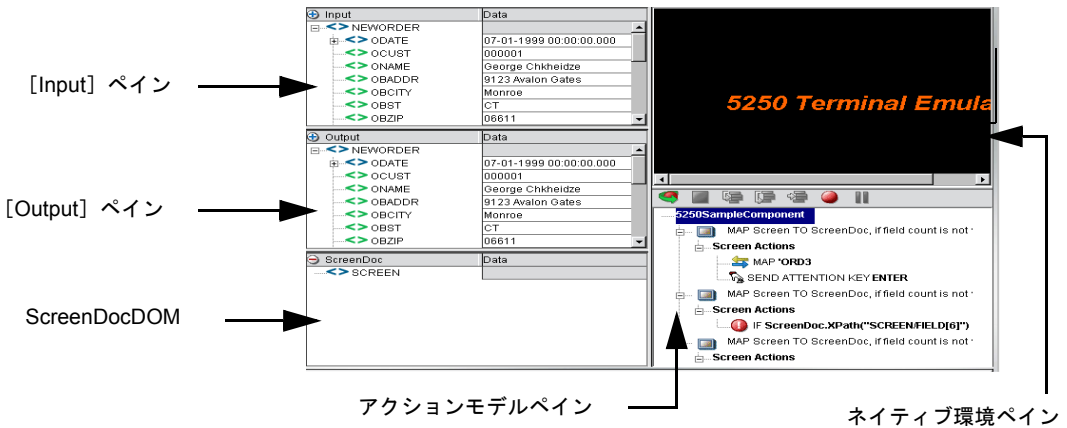
- 8** プルダウンリストで「**接続**」の名前を選択します。5250 接続の詳細については、18 ページ「5250 接続リソースの作成」を参照してください。
- 9** [**Finish**] をクリックします。コンポーネントが作成され、5250 コンポーネントエディタが表示されます。



5250 コンポーネントエディタウィンドウについて

5250 コンポーネントエディタには、XML Map コンポーネントエディタの機能がすべて含まれています。また、入力 XML ドキュメントと出力 XML ドキュメントのマッピングペインや、アクションペインも含まれています。

ただし、従来の機能との大きな違いが 2 つあります。1 つ目は、5250 コンポーネントエディタにはすべての Connect に共通するネイティブ環境ペインも含まれているという点です。このペインには、5250 エミュレータが搭載されています。また、5250 コンポーネントエディタの [Record] ボタンを選択するまで、このペインは空白で表示されます。[Record] ボタンを押すと、5250 コンポーネントで使用される接続で指定したホストとの 5250 エミュレーションセッションが、ネイティブ環境ペイン内で確立されます。2 つ目は、ScreenDoc という名前の XML DOM がコンポーネントエディタウィンドウに追加されているという点です。この DOM は、ホストから受け取った各画面の XML ドキュメント表現を表します。また、この DOM は、コンポーネント内の参照やマッピングアクションの作成に使用したり、ユーザが画面フィールドを簡単に参照できるように式ビルダで使用したりすることが可能です。5250 画面のクイック HTML 表現を作成する場合は、ScreenDoc DOM の出力を使用したり、スタイルシートを適用して画面の HTML 表現を作成したりすることが可能です。



5250 ネイティブ環境ペインについて

5250 ネイティブ環境ペインには、AS/400 環境の 5250 エミュレーションが装備されています。このペインでは、次のことを実行できます。

- ◆ 入力XMLドキュメント(または他の使用可能なDOM)からデータをマップして、5250 画面フィールドの入力として使用する。たとえば、SKU 番号を入力DOM から 5250 画面のパーツフィールドにドラッグして、ホストを照会し、そのパーツ番号に関連付けられているデータ(説明や価格など)を返すことができます。
- ◆ 返された5250画面からデータをマップして、出力XMLドキュメント(または、Temp、MyDom などの他の使用可能なDOM)に配置する。
- ◆ ヘッダ情報および詳細情報(複数の品目から成る請求書など)を、特別なMulti-Row アクションを使用して、XML ドキュメントからネイティブ環境ペインでアクセスされるトランザクションにマップする。
- ◆ ヘッダ情報および詳細情報(カスタマ名やトランザクション履歴など)を、ネイティブ環境ペインのトランザクションからXMLドキュメントにマップする。

ネイティブ環境ペインのトランザクション機能は、5250 端末や端末エミュレータのトランザクション機能と同じです。

5250 キーボードサポートについて

5250 ネイティブ環境ペインでは、アテンション ID (AID) キー (Enter、Clear、PA1-3、および PF1-24) の使用がいくつかサポートされています。各アテンションキーの機能は、ホストアプリケーションにより異なります。これらのキーは、PC キーボードにマップされます (次を参照)。

表 3-1

5250 キー	PC キー
Enter	Enter
Clear	ESC
F1 から F12	F1 から F12
F13 から F24	Shift+F1 から Shift+F12
PA1 から PA3	Ctrl+F1 から Ctrl+F3

5250 コンポーネントを作成する際、キーボードのキーを直接使用するか、[View] メニューのキーボードツールバーを使用できます。

ScreenDoc DOM について

ScreenDoc DOM は、ネイティブ環境ペインの端末データストリームを介して受け取った現在の画面の XML ドキュメント表現です。この画面に対するすべてのマップアクション (ドラッグアンドドロップなど) は、実際には ScreenDoc DOM の要素を参照します。任意のアプリケーション画面を XML ドキュメントとして扱うと同時に、表示および参照できます。

機能

5250 コンポーネントは、非同期でブロックモードの端末データストリームを介してホスト環境との通信を行います。データのブロックは、基本的に画面を表します。ホストでは、コンポーネントに表示される画面ブロックを送信します。この画面は、ユーザ (最終的にはユーザの作成するコンポーネント) により編集されます。編集された画面は、AID キーを押した後の処理のためにホストに送り返されます。

動作

記録モード中、データの画面ブロックがコンポーネントにより受信されるたびに、次の 4 つのことが同時に起こります。

- ◆ 新しい画面がネイティブ環境ペインに表示される。

- ◆ **Map Screen** アクションがアクションモデルペインに表示される。Map Screen アクションでは、この特定の画面のアクションを追加、変更、および削除します。新しい画面ブロックがコンポーネントの作成時に受信されるたびに、新しい **Map Screen** アクションが作成されます。
- ◆ **Map Screen** アクションでは、[TDS] フィールドがこの特定の画面に対して何回受信されたかを計算および記録する。この情報は、次にコンポーネントを実行するときに、検証の目的で使用されます。
- ◆ **ScreenDoc DOM** が、受信した画面を反映する新しい **DOM** で更新される。ブロックモード端末では、フィールドのストリームとしてデータを送信します。これらのフィールドは、ホスト環境 (CICS の BMS など) で画面作成ユーティリティを使用して定義されます。

フィールドは、**ScreenDoc DOM** で表示されます。表示は、画面の左上角から始まり、右に向かいます。行末に達すると、1 行下がり、左端からまた続きます。これは、全体の 80x24 または 132x27 の画面領域に達するまで続きます。元の画面レイアウトの定義に応じて、複数のフィールドが存在することもあります。いくつかのフィールドは、画面のテキストラベルで、通常は **protected (prot)** および **bright (brt)** という表示属性を持ちます。また、データエントリ用のフィールドもあり、これらのフィールドの属性は **unprotected (unprot)** です。さらに、データを含むが表示されないフィールドもあり、これらの属性は **dark (drk)** です。そして、画面のタブ機能を実装する特殊な画面フィールドもあります。このフィールドではデータを入力できず、属性が **bypass** のためバイパスフィールドと呼ばれます。

すべての **TDS** フィールドをリストすると、**ScreenDoc DOM** が非常に大きくなる可能性があります。すべてのフィールドのリストは、主に、非表示フィールドの検索、属性によるフィールドおよびその位置の確認、および便利な場合は **Composer** のドラッグアンドドロップ機能を使用した **ScreenDoc DOM** から出力 **DOM** へのマップに使用されます。

注記： 通常、**ScreenDoc DOM** にマップするよりも、ドラッグアンドドロップを使用してネイティブ環境ペインに対して直接マップした方が、より簡単で効率的です。

TDS の各フィールドは、**FEILD** という名前の要素として **ScreenDoc DOM** に表示されます。**ScreenDoc DOM** には、**TDS** のフィールドと同じ数の **FIELD** 要素が含まれます。この要素では、フィールドに定義されているデータを表示します。画面フィールドは、フィールドラベルおよびデータエントリフィールドの両方で使用されます。また、各要素により、フィールドの次の属性が表示されます。

表 3-2

属性	意味
Column	画面列 (1 から 80)。フィールドは左から始まります。
Display	TDS のフィールドに定義される表示属性 (prot = protected、brt = bright、unprot= unprotected、dark = dark、bypass = bypass)

属性	意味
Id	TDS でのフィールド位置を表す絶対シーケンス番号
Length	フィールドの固定長
Row	画面行 (1 から 24)。フィールドは上から始まります。

5250 固有のメニューバー項目について

[View] メニュー

[**Keypad Tool Bar**] - 5250 端末キーのキーパッドツールバーを表示します。このツールバーは、ネイティブ環境ペインの上部にドッキングされます。テンキーは、この位置からドラッグして、フローティングウィンドウにすることができます。フローティングウィンドウを閉じると、テンキーは、ネイティブ環境ペインに戻ります。テンキーを非表示にする場合、メニューバーから [**View**]、[**Keypad Tool bar**] の順に選択します。

[Component] メニュー

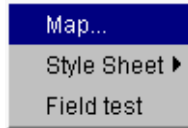
[**Style Sheet**] - スタイルシートエディタダイアログボックスを表示します。このダイアログボックスには、Composer メインウィンドウのリソースとして表示される事前に定義されたスタイルシートがいくつか含まれます。

[**Start/Stop Recording**] - 画面トランザクションを操作する場合にアクションの自動作成を管理します。[**Start**] は、画面を操作するときにアクションを作成し、[**Stop**] は、アクション作成を終了します。

[**Connect/Disconnect**] - ホストへの接続を制御できます。記録またはアニメーション表示を行う場合、接続は自動的に確立されるので、このボタンを使用する必要はありません。ただし、このボタンは、記録またはアニメーション表示を行わずに、単に5250環境を移動することを目的として接続を確立したい場合に便利です。

5250 固有のコンテキストメニュー項目について

5250 Connect には、この Connect に固有なコンテキストメニュー項目も含まれています。コンテキストメニューを表示するには、ネイティブ環境ペインにカーソルを置き、マウスを右クリックします。コンテキストメニューが表示されます(次を参照)。

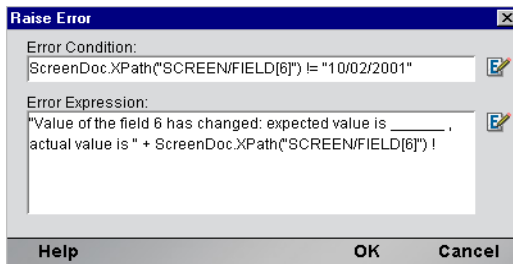


コンテキストメニュー項目の機能は、次のとおりです。

[Map] - Map アクションを作成できます。Map アクションを作成するには、入力 DOM でソースを選択し、続けてネイティブ環境ペインでソースを選択します。この結果、Map アクションが作成されます。また、ネイティブ環境ペインでマウスを右クリックし、[Map] を選択しても、アクションを作成できます。

[Style Sheet] - 別のスタイルシートを適用して、ネイティブ環境ペインの外観を変更できます。

[Field Test] - 選択したフィールドの Raise Error アクションを作成できます。マウスを右クリックして、[Raise Error] を選択すると、式が自動的に入力され、アクションが作成されます。この例については、次の画面を参照してください。



5250 固有のボタンについて

5250 Connect では、コンポーネントエディタツールバーに、[Record] ボタンおよび [Connect/Disconnect] ボタンという追加のツールボタンが 2 つあります。[Record] ボタンを使用すると、画面トランザクションを操作するときに、アクションモデルにアクションが自動的に作成されます。[Connect/Disconnect] ボタンを使用すると、ホストへの接続を制御できます。これらのボタンは、次のとおりです。

オフ



オン



接続



切断



[Record]

[Connect/Disconnect]

4

5250 アクションの実行

アクションについて

「アクション」は、プログラミングステートメントに類似しており、パラメータの形式で入力を受け付け、特定のタスクを実行します。『Composer ユーザガイド』のアクションに関する章を参照してください。

5250 コンポーネントエディタ内では、XML ドキュメントを処理したり、非 XML データソースと通信したりするための命令のセットが、「アクションモデル」の一部として作成されます。アクションモデルは、AS/400 と XML ドキュメント間でのすべてのデータマッピング、データ変換、データ転送、およびコンポーネントとサービス内でのデータ転送を実行します。

アクションモデルは、アクションのリストから構成されています。アクションモデル内のすべてのアクションは相互に機能します。たとえば、あるアクションモデルでは、請求書のデータをディスクから読み取り、データを AS/400 のインベントリデータベースから取得し、一時 XML ドキュメントに結果をマップして変換し、変換されたデータを出力 XML ドキュメントにマップします。

このアクションモデルの例は、いくつかのアクションから構成されています。そのアクションは次のとおりです。

- ◆ 請求書のドキュメントを開き、5250 コマンドを実行して AS/400 のデータベースから請求書のデータを取得する
- ◆ 結果を一時 XML ドキュメントにマップする
- ◆ コードテーブルを使用して数値コードを変換し、結果を出力 XML ドキュメントにマップする

5250 専用アクションについて

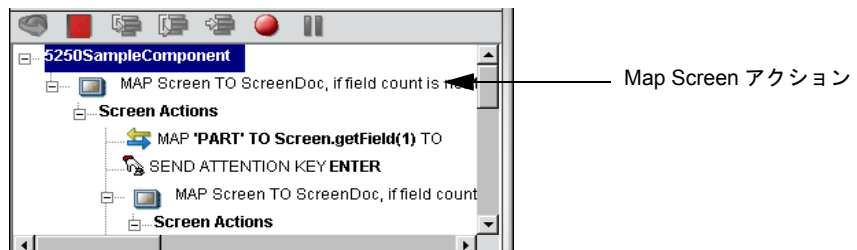
5250 コネクタには、5250 専用のアクションがいくつか含まれています。これらのアクションは Composer には含まれていません。

表 4-1

5250 アクション	説明
Map Screen	特定のトランザクション画面に関連したアクションを配置する、アクションモデル内の場所を示します。
Multi Row	このアクションでは、DOM と 5250 画面の間で、複数対複数のデータ関係のマップを指定することができます。
Send Attention Key	このアクションは、Aid キーを押すと、自動的に作成されます。このアクションを編集して、ホストに返信するキーを変更できます。

Map Screen

Map Screen アクションは、特定のトランザクション画面が表示されるアクションモデル内の場所を示すだけでなく、エラーのチェックに使用することもできます。画面が最初に記録されると、Composer により、画面のフィールド数が保存されます。この数は後で、記録されたアクションが適切な情報を返すかどうかを確認するためのエラーチェックの実行中に比較されます。アクションモデルに Map Screen アクションが表示されます (次の図を参照)。

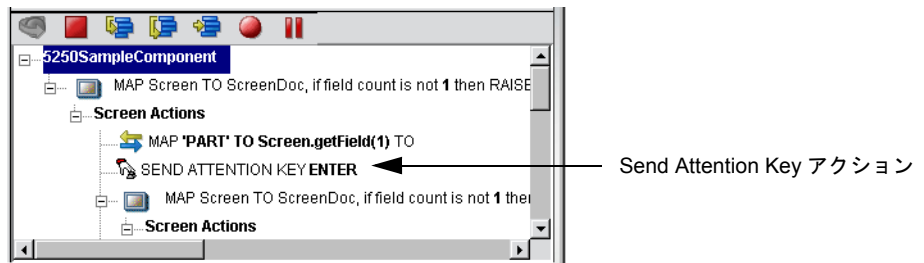


Multi Row

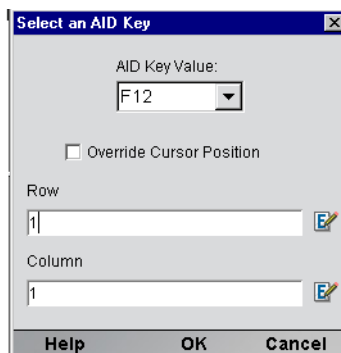
Multi Row アクションは、XML ドキュメントのデータを 5250 画面に入力したり、5250 画面のデータを XML ドキュメントに出力したりするために使用できます。基本的にこのアクションは、アクションモデル内に繰り返ループを作成します。これにより、データの複数の行が、あるドキュメントまたは画面から別のドキュメントまたは画面に自動的にマップされます。Multi Row アクションの詳細については、59 ページ「5250 Multi Row ウィザードについて」で説明されています。

Send Attention Key

ネイティブ環境ペインのツールバーに表示される AID キーの 1 つまたはそれに対応するキーボードキー、またはキー (36 ページ「5250 キーボードサポートについて」を参照) を選択すると、アクションモデルに Send Attention Key アクションがマップされます。



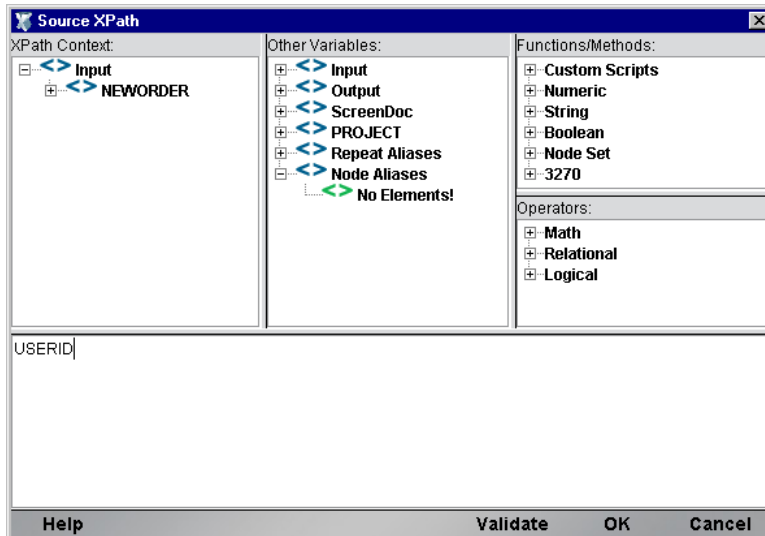
アクションモデルで Send Attention Key アクションをダブルクリックすると、キーを編集できるダイアログボックスが表示されます。ドロップダウンリストボックスから、値キーを選択します。カーソル位置を上書きするためのチェックボックスをオンにします。必要に応じて行と列を編集するか、Expression Builder のアイコンをクリックして、計算を追加します。



5250 専用の Expression Builder 拡張

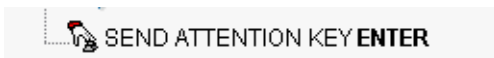
TN5250 接続リソースには、アクションの Expression Builder のダイアログボックスからアクセスできる、「ユーザ ID」と「パスワード」という 2 種類のアイテムがあります。これらは、接続が確立されたときに Composer 画面に最初に表示される、[UserID] と [Password] というフィールドで使用されます。これらの変数を画面にマップすると、画面にマップするための Map アクションに変数を入力する必要がなくなります。

ソースの式が PASSWORD、ターゲット XPath が \$SCREENDOC/FIELD(5) と指定されている Map アクションを作成できます。



5250 セッションの記録

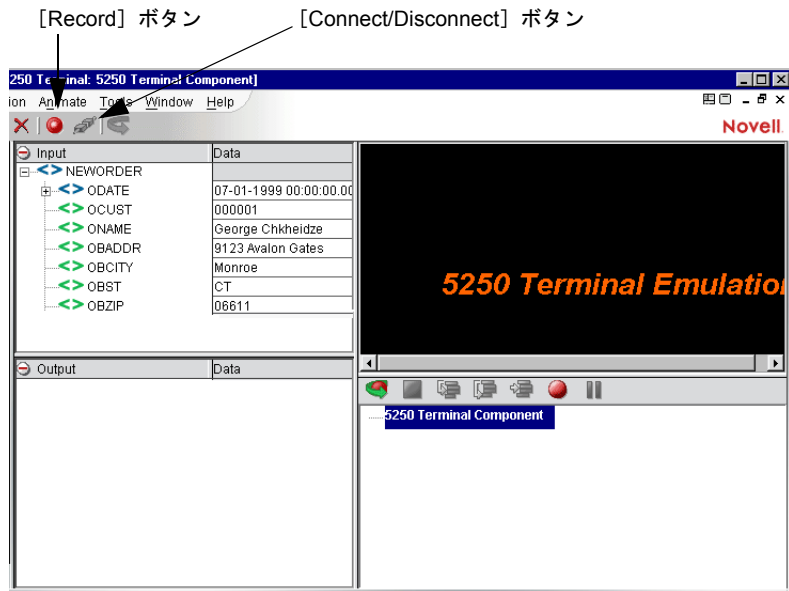
5250 コンポーネントは、すべての UI コンポーネントと同様、アクションモデルの主な部分 (ナビゲーション部分) が自動的に作成されるため、他のコンポーネントとは異なります。これは、ネイティブ環境ペインでホストからのライブセッションと通信し、Composer がアクティビティをアクションのセットとしてアクションモデルに記録するためです。他のコンポーネントでは、ユーザは、マップ、変換、およびタスクの転送を実行するためのアクションを、アクションモデルで手動で作成する必要があります。5250 コンポーネントを作成では、基本的に AS/400 間の要求と応答の通信を記録します。これにより、コンポーネントの作成に必要なキー操作と画面移動を指定するアクションがアクションモデルに生成されます。たとえば、5250 のネイティブ環境ペインで [Enter] ボタンを選択すると、次の図に表示されるように、アクションモデルにアクションが記録されます。また、他のコンポーネントと同様に、アクションモデルにアクションを追加することもできます。



注記： 5250 コンポーネントを適切に作成するには、5250 のコマンドと、XML 統合プロジェクトで使用するアプリケーションについて精通している必要があります。

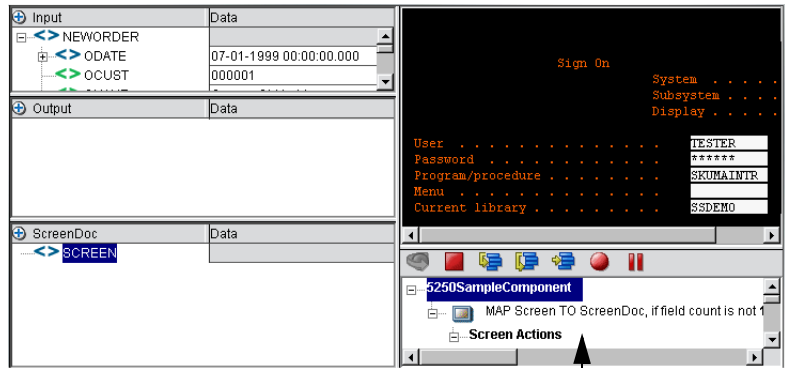
➤ 5250 セッションを記録する

- 1 31 ページ「5250 コンポーネントを作成する前に」の指示に従って、5250 コンポーネントを作成します。新しく作成した 5250 コンポーネントは、5250 コンポーネントエディタウィンドウに表示されます。



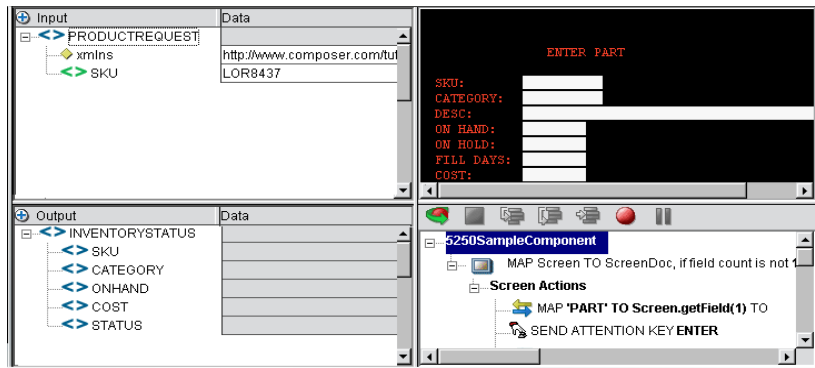
注記： 5250 コンポーネントエディタには、XML Map コンポーネントエディタのツールバーに表示されるボタンの他に、**[Record]** ボタンと **[Connect/Disconnect]** ボタンがあります (図を参照)。

- 2** **[Record]** ボタンをクリックします。ネイティブ環境ペインに入力画面が表示され、アクションモデルペインに、「Map Screen」アクションが記録されます。

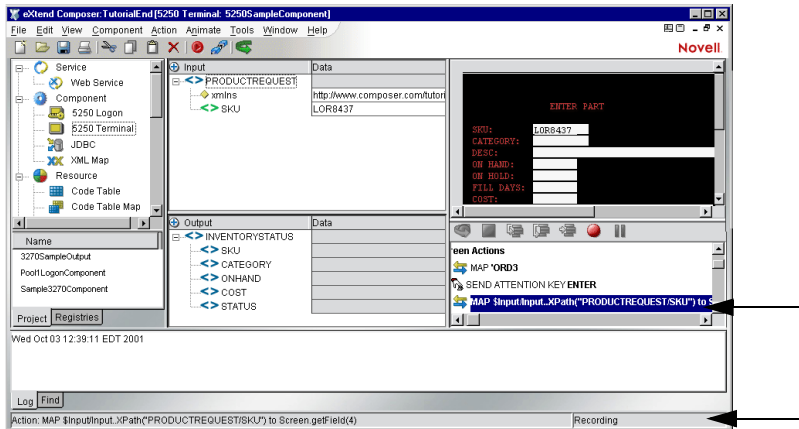


記録されたアクション

- 3 [UserID] にユーザ ID、[Password] にパスワード、およびその他の必要な情報を入力します。この例では、[UserID]、[Password]、[Program/Procedure]、[Current Library] のエントリを入力する必要があります。
- 4 キーボードで、<Enter> キーを押すか、ツールバーが表示されている場合 (表示するには [View]、[Keypad Toolbar] の順にクリック)、[Enter] ボタンをクリックします。この例では、5250 画面に [ENTER PART] 画面が表示されます。



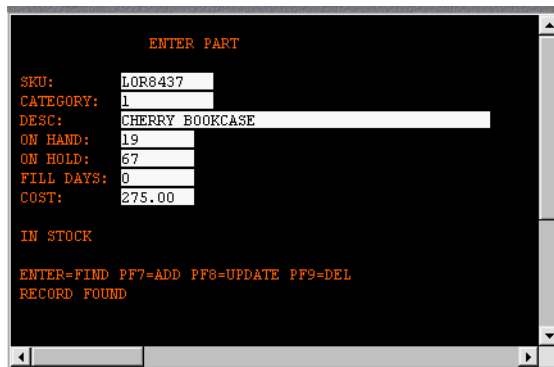
- 5 入力 DOM から 5250 の [ENTER PART] 画面の [SKU] フィールドに、SKU データをドラッグします。アクションは、アクションモデルペインに記録され、ステータスバーに表示されます。



ドラッグアンドドロップアクションの結果を反映したアクションモデルとステータスバー

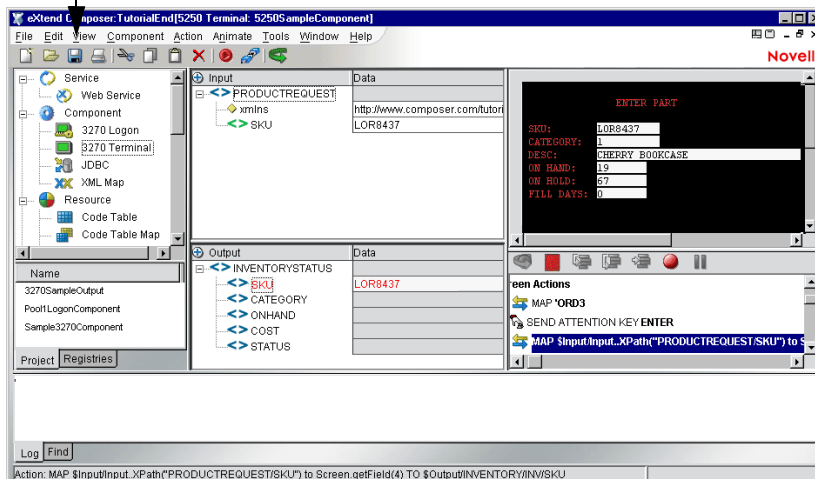
注記： Map アクションを使用して、入力 SKU を [ENTER PART] 画面の [SKU] フィールドにマップすることもできますが、ドラッグアンドドロップの方がはるかに迅速に操作を行うことができます。Map アクションの詳細については、『eXtend Composer ユーザガイド』を参照してください。

- 6 5250 のペインで [Enter] ボタンをクリックします。5250 の [ENTER PART] 画面には、SKU に関連付けられたデータが表示されます。



- 7 [ENTER PART]画面から出力 DOM に、要素(たとえば、SKU 番号)をドラッグアンドドロップします。ドラッグアンドドロップしたデータは出力 DOM に赤で表示されます。

[Save] ボタン



- 8 データ要素を [ENTER PART] 画面から出力 DOM の該当するフィールドにドラッグアンドドロップする操作を必要に応じて繰り返します。要素を [ENTER PART] 画面から出力 DOM にドラッグするたびに、アクションモデルペインにアクションが記録されます。
- 9 [Save] ボタンをクリックします。

以前に記録したアクションモデルの編集

以前に記録したアクションモデルの編集が必要となる場合が必ずあります。他のコンポーネントの編集とは異なり、5250 コンポーネントの編集には、さらに注意が必要です。5250 コンポーネントを実行すると、コンポーネントが適切に動作するために、特定の画面およびデータが表示されるようなアクションのシーケンスが繰り返されます。そのため、コンポーネントを編集する際には、アクションモデルのシーケンスが以前に記録したホストトランザクションの実行シーケンスと矛盾しないように注意する必要があります。

一般的に、正常に編集を行うには、次の推奨事項が適用されます。

- ◆ 「Map Screen」アクションのブロックを切り取ってコピーしたり、アクションモデル内の別の場所に貼り付けたりしないでください。
- ◆ アクションモデル内で、画面と通信する個々の Map アクションをコピーしたり貼り付けたりした後は、各アクションのチェックと編集を慎重に行ってください。

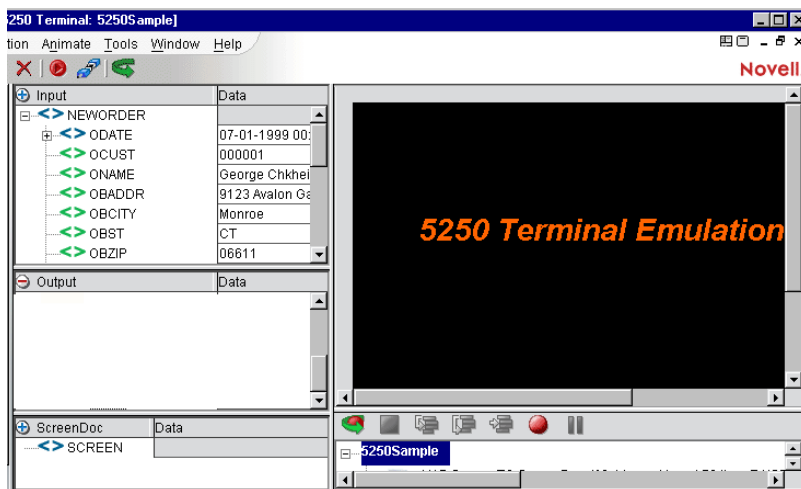
- ◆ Composer のドラッグアンドドロップ機能を使用して、画面と動作する新しい Map アクションを追加します。アクションモデルで、所定の行までアニメーション表示した後、一時停止し、記録モードに戻ります。これにより、アクションモデルが適切な ScreenDoc DOM または特定の ScreenDoc DOM 内のフィールド、あるいはその両方と同期が取れなくなることを防ぐことができます。
- ◆ アニメーション表示中は、アクションモデル内の Multi-Row に関する行 (またはアクション) を削除しないでください。それらを削除すると、コンポーネントが適切に動作しなくなる場合があります。

既存のアクションの変更

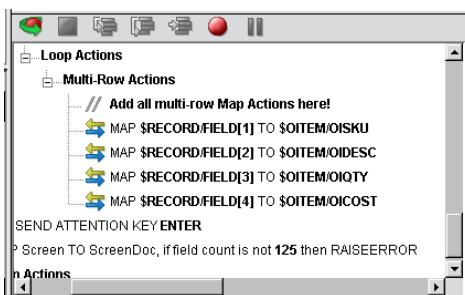
次の手順では、以前に記録したセッションで既存のアクションを変更する方法を説明します。

➤ 以前に記録されたアクションモデルで、既存のアクションを変更する

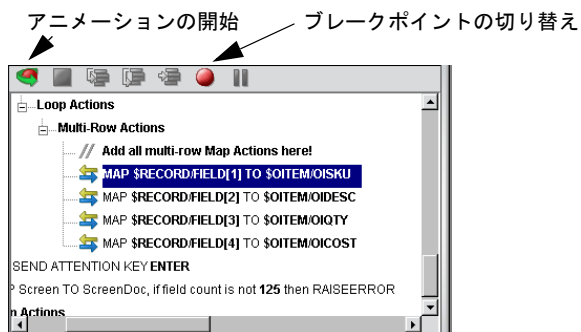
- 1 編集する対象となる、以前に記録されたアクションモデルを含むコンポーネントを開きます。5250 コンポーネントエディタウィンドウにコンポーネントが表示されます。



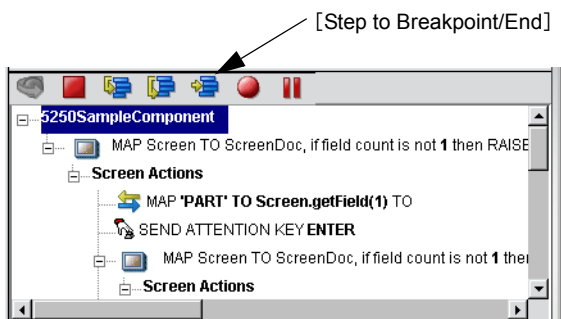
- 2 アクションモデルで、編集を行うアクションに移動し、選択します。



- 3 [Toggle Breakpoint] ボタンをクリックします。選択したアクションが赤色になります。

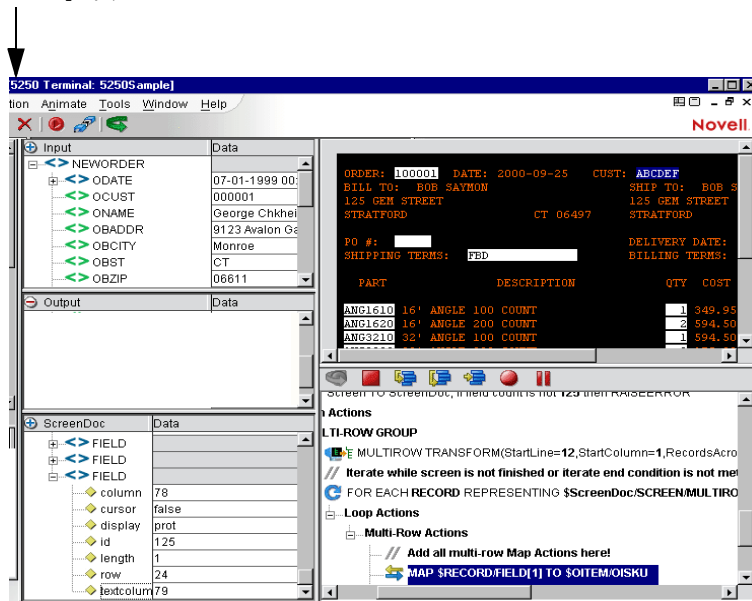


- 4 [Start Animation] ボタンをクリックします。アニメーションツールが有効になります。



- 5 [Step to Breakpoint/End] ボタンをクリックします。アクションモデルは、最初から手順 3 で設定したブレーポイントまでのすべてのアクションを実行します (次の図を参照)。

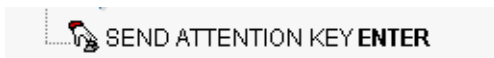
[Record] ボタン



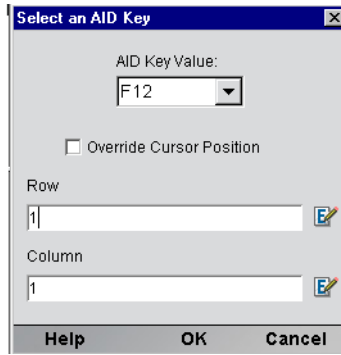
- 6 アニメーションツールバーで、[Pause] ボタンをクリックします。
- 7 コンポーネントエディタのツールバーで、[Record] ボタンをクリックします。
- 8 アクションモデルに追加するアクションを実行します。
- 9 [File]、[Save] の順に選択するか、コンポーネントエディタのツールバーで [Save] ボタンをクリックします。
- 10 81 ページ「アニメーションツールの使用」の指示に従って、コンポーネントをテストします。

アテンションキーの編集

セッションの記録中に、キーボードで <Enter> キーを押すか、5250 のネイティブ環境ペインで多数のアテンションキーのうちのいずれかをクリックすると、アクションモデルにアクションが記録されます。例は次のとおりです。



これらのアクションは、他のアクションと同様に、削除、移動、またはコピーすることが可能ですが、移動を伴うため、コンポーネントの実行に影響を及ぼすことに注意してください。Attention Key アクションをダブルクリックし、表示されるダイアログボックスで編集することもできます。

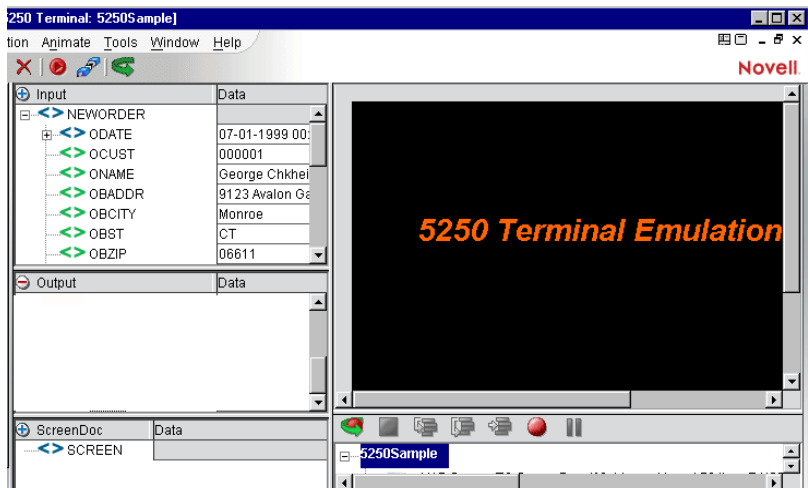


新しいアクションの追加

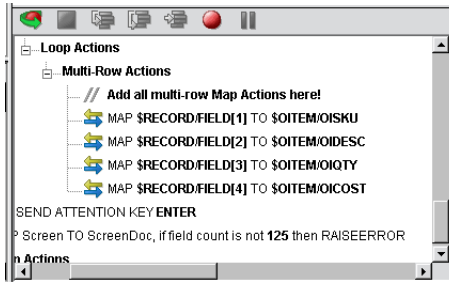
次の手順では、以前に記録したセッションで新しいアクションを追加する方法を説明します。

➤ 新しいアクションを以前に記録したアクションモデルに追加する

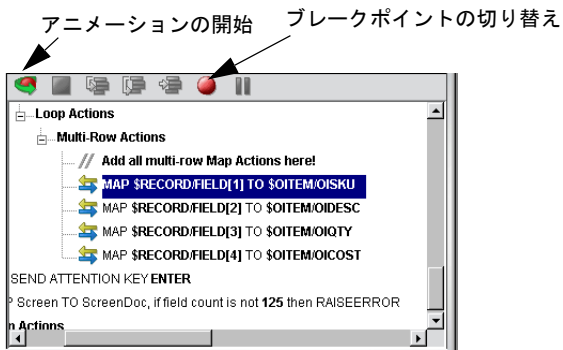
- 1 アクションを追加する対象となる、以前に記録したアクションモデルを含むコンポーネントを開きます。5250 コンポーネントエディタウィンドウにコンポーネントが表示されます。



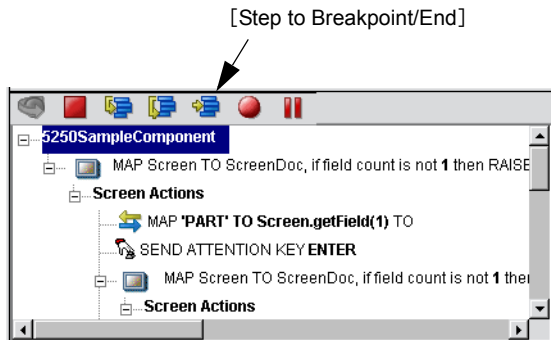
- 2 アクションモデルで、追加するアクションに移動し、選択します。



- 3 [Toggle Breakpoint] ボタンをクリックします。選択したアクションが赤色になります。

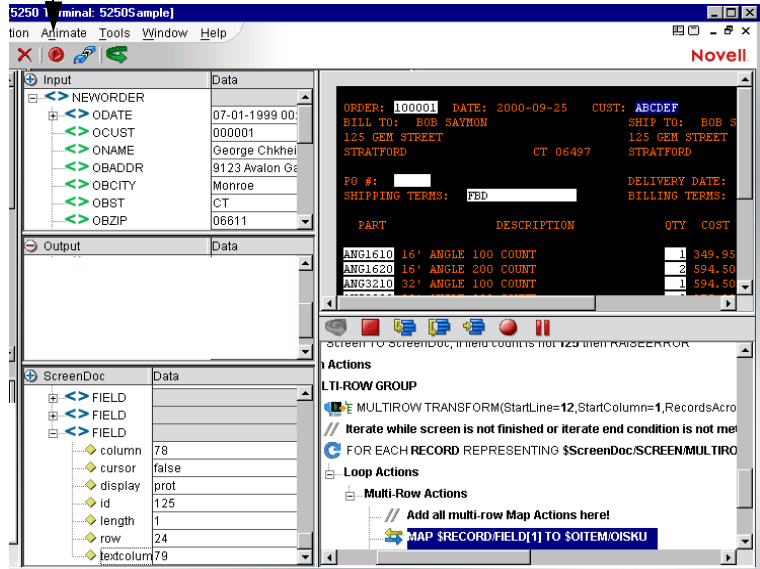


- 4 [Start Animation] ボタンをクリックします。アニメーションツールが有効になります。



- 5 [Step to Breakpoint/End] ボタンをクリックします。アクションモデルは、最初から手順3で設定したブレーポイントまでのすべてのアクションを実行します(次の図を参照)。

[Record] ボタン



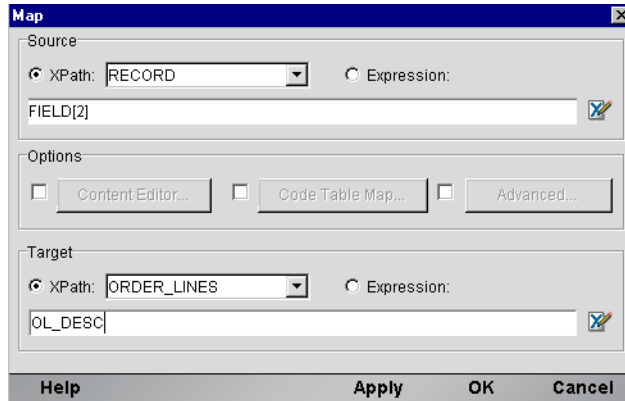
- 6 アニメーションツールバーで、[Pause] ボタンをクリックします。
- 7 コンポーネントエディタのツールバーで、[Record] ボタンをクリックします。
- 8 Composer のドラッグアンドドロップ機能を使用して、画面と通信する新しい Map アクションを追加します。選択した行のすぐ下に新しいアクションが追加されます。
- 9 [File]、[Save] の順に選択するか、コンポーネントエディタのツールバーで [Save] ボタンをクリックします。
- 10 81 ページ「アニメーションツールの使用」の指示に従って、コンポーネントをテストします。

Alias アクションの追加について

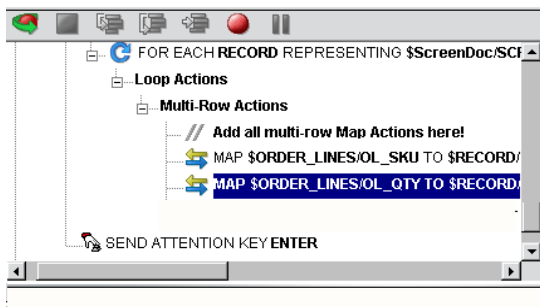
別名であるループに Map アクションを追加している場合、次の操作を実行します。

- Alias アクションを以前に記録したアクションモデルに追加する
- 1 コンポーネントを開きます。

- 2 [Action] メニューから、[New Action/Advanced]、[Map] の順に選択します。
[Map] ダイアログボックスが表示されます。



- 3 [Source] のドロップダウンリストから XPath を選択します。ドロップダウンリストから [Order_Lines] が選択されます。
- 4 情報を入力するか、[Expression Builder] ボタンをクリックして新しい式を作成します。
- 5 別名によって表される XPath を作成します。ドロップダウンリストから別名をクリックします。
- 6 [OK] をクリックします。
- 7 選択した行の下に新しいアクションが挿入されます。(次の図では、挿入された新しい行がハイライト表示されています)。

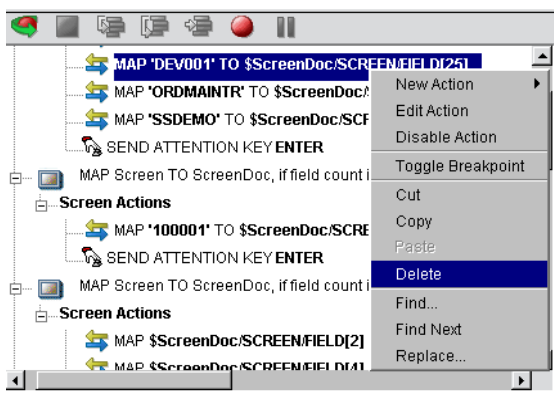


アクションの削除

次の手順では、以前に記録したセッションでアクションを削除する方法を説明します。

➤ 以前に記録したアクションモデルでアクションを削除する

- 1 削除するアクションの行を選択して、マウスを右クリックし、メニューから [Delete] を選択します。行を選択して、キーボードの <Delete> ボタンを押すこともできます。



5250 Multi Row ウィザードについて

5250 Connect には、データを XML ドキュメントから 5250 へまたは 5250 画面から XML ドキュメントへ迅速で簡単にドラッグすることのできる、ドラッグアンドドロップ機能があります。ただし、一定でない行数や、複数の画面に渡るデータがある場合、ドラッグアンドドロップは有効な方法ではありません。この問題を解決するため、5250 Connect には、DOM と 5250 画面の間で複数対複数のデータ関係のマップを指定するための「Multi Row ウィザード」があります。Multi Row ウィザードは、5250 コンポーネントで [Action] > [New Action] の順に選択すると表示されるオプションです。これは、データを XML ドキュメントから 5250 画面の繰り返しの領域に入力したり、繰り返されるデータを 5250 画面から XML ドキュメントに出力したりする場合に使用します。基本的に Multi Row ウィザードによって、データの複数行をあるドキュメントまたは画面から別のドキュメントまたは画面に、完了するまで自動的にマップするアクションモデル内に、行数や画面数に関わらず、繰り返しループが作成されます。

このガイドで使用する Multi Row サンプルについて

Multi Row ウィザードを使用して複数行のデータを XML ドキュメントおよび 5250 画面に入出力するための方法はたくさんありますが、このガイドではすべてのシナリオには言及していないことをご理解ください。ここでは、Multi Row ウィザードの一般的な使用方法を理解するための、基本となる 2 種類のサンプルコンポーネントを紹介します。

最初の手順では、複数のデータ要素を XML ドキュメントから 5250 画面に「入力」するための Multi Row ウィザードを使用するコンポーネントを示します。2 つ目の手順では、複数のデータ要素を 5250 画面から出力 DOM に「出力」するための Multi Row ウィザードを使用するコンポーネントを示します。

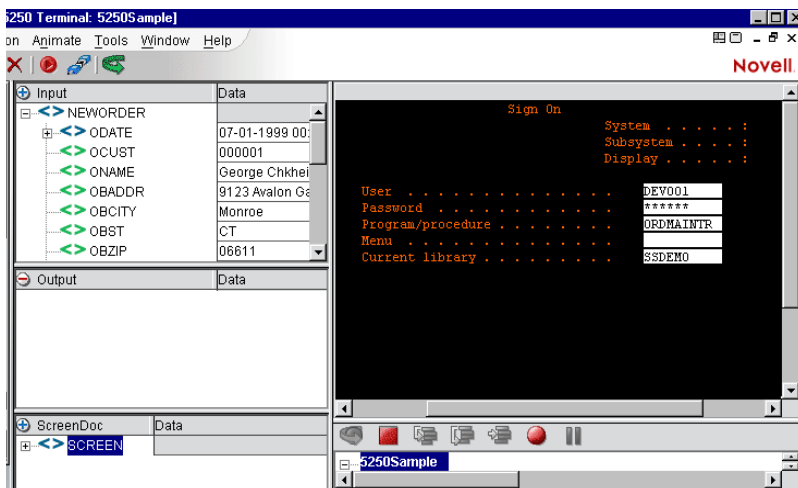
この節で扱う手順は、単に例として示すものです。Compoer のチュートリアルとは異なり、これらの手順は段階に沿って使用できるものではありません。

➤ Multi Row ウィザードを使用したデータの入力を準備する

- 1 31 ページ「5250 コンポーネントを作成する前に」の指示に従って、コンポーネントを作成します。この例では、コンポーネントを「5250SampleInput」と呼びます。次の図の 5250 コンポーネントエディタに、コンポーネントを示します。



- 2 [Record] ボタンをクリックします。ネイティブ環境ペインにサインオン画面が表示されます。

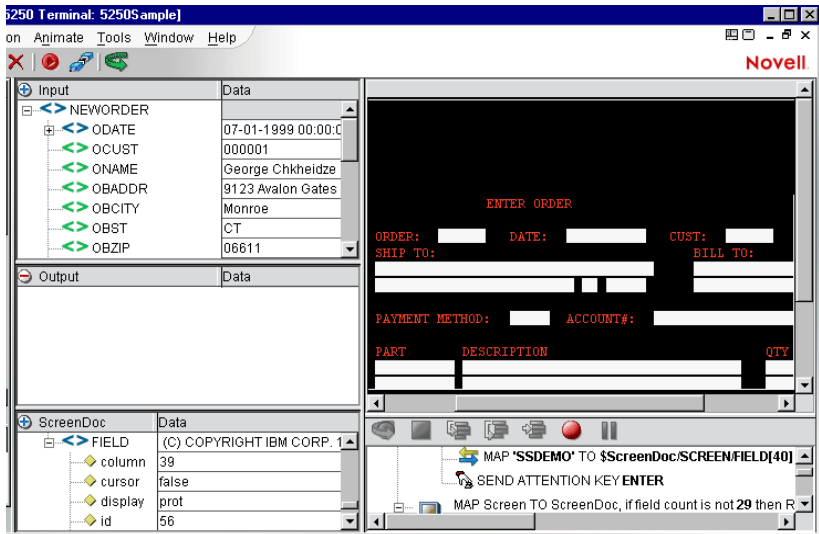


- 3 この例で使用する適切な画面に移動し、[UserID]、[Password]、および [Program/Procedure] のエントリを入力します。この例では、プログラム/プロセスとして「GORD」を使用しています。

- 4 キーボードで、<Enter>を押します。

注記： また、ネイティブ環境ペイン内で、5250 のツールバーから、[Enter] を選択することもできます。このツールバーを表示するには、5250 コンポーネントエディタのメニューバーから [View]、[Keypad Toolbar] の順に選択します。

この例では、ネイティブ環境ペインに、「ENTER ORDER」という 5250 画面が表示されます。



- 5 次に、入力 DOM から、5250 の [ENTER ORDER] 画面内の対応するフィールドに、該当するヘッダデータをドラッグアンドドロップします。この例では、ドラッグアンドドロップされるデータは、次のとおりです。

表 4-2

マップ元: DOM フィールド	マップ先: 5250 の [ENTER ORDER] 画面のフィールド
ODATE	DATE:
OCUST	CUST:
ONAME	NAME:
OBADDR	BILL TO: (1 行目)
OBCITY	BILL TO: (2 行目)
OBST	BILL TO: (2 行目、都市名の右側にある 2 文字用のフィールド)
OBZIP	BILL TO: (2 行目、州名の右側にある 5 文字用のフィールド)
OSADDR	SHIP TO: (1 行目)
OSCITY	SHIP TO: (2 行目)
OSST	SHIP TO: (2 行目、都市名の右側にある 2 文字用のフィールド)
OSZIP	SHIP TO: (2 行目、州名の右側にある 5 文字用のフィールド)
OPAY	PAYMENT METHOD:

マップ元: DOM フィールド マップ先: 5250 の [ENTER ORDER] 画面のフィールド

OPAYINFO

ACCOUNT #:

5250 の [ENTER ORDER] 画面が次のように表示されます。

The screenshot shows a terminal window titled "ENTER ORDER". The fields are as follows:

ORDER:		DATE:	07-01-1999	CUST:	000001	NAME:	Chris Keller
SHIP TO:				BILL TO:			
60 Crown View Drive				60 Crown View Drive			
Monroe CT 06468				Monroe CT 06468			
PAYMENT METHOD:	VISA	ACCOUNT#:	4111-1111-1111-1111				

PART	DESCRIPTION	QTY	COST	EXT

フィールドを編集する場合、次の手順に従います。

注記： 5250 画面は、24 行 X80 列で構成されています。

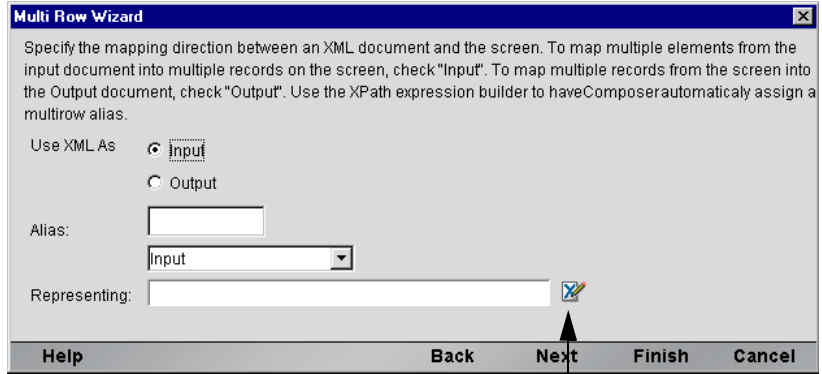
- ◆ [Start Line] - ウィザードがカウントを開始する最初の行です。
- ◆ [Start Column] - ウィザードを開始する最初の列です。Column 1 は一番左の列です。
- ◆ [# Records Down] - ウィザードを使用してループに含めるレコードの数です。
- ◆ [# Records Across] - レコードをページの一番下までリストし、次に折り返して最上部に戻るのが、COBOL のプログラムの一般的な方法です。これにより、ページの下に向かって、いくつかのレコードが隣り合って表示されます。このフィールドを使用して、1 つの行内に並んで表示されるレコードの数を指定します。
- ◆ [# Lines/Record] - レコードが 80 文字を超える場合、レコードは次の行に折り返されます。レコードが 160 文字を超える場合、レコードは 3 行目まで折り返されます。1 つのレコードに使用する行数を指定する必要があります。
- ◆ [# Columns/Record] - このフィールドでは、レコードに含める列数を指定します。最大列数は 80 です。

- 2 必要に応じてフィールドを編集します。パラメータが適切である場合、[Next] をクリックします。Multi Row ウィザードの 2 番目のパネルが表示されます。

➤ Multi Row ウィザードに Repeat アクションを設定する

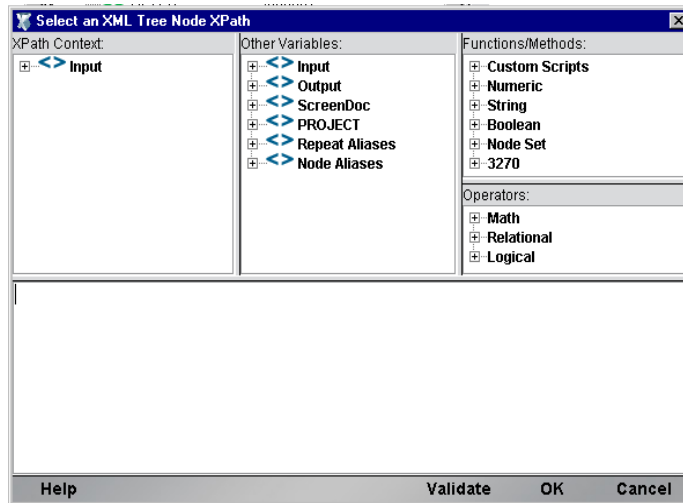
注記： データを出力する場合、71 ページ「Multi Row ウィザードを使用してデータを出力する」を参照してください。

- 1 Multi Row ウィザードの最初のパネルが完了すると、次のダイアログボックスが表示されます。



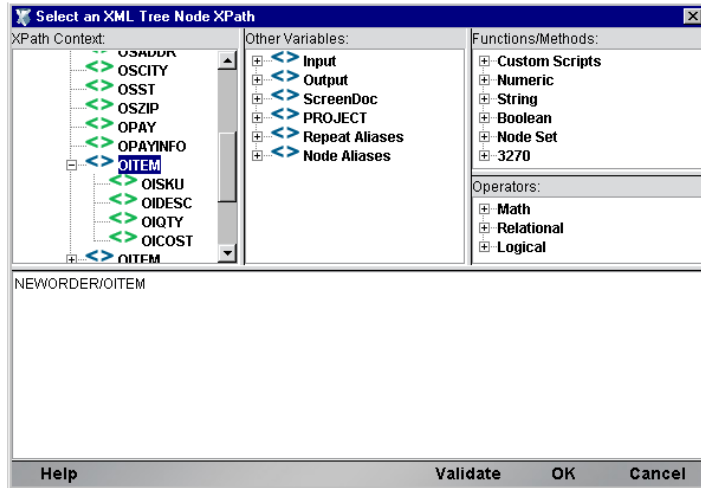
Expression Builder

- 2 [Use XML As] で、[Input] ラジオボタンを選択します。このパネルは、複数の要素または画面の行を処理する Repeat アクションを作成するために使用します。これは、すべてのコンポーネントで使用できる基本の「Repeat for Element」アクションと類似したアクションです。
- 3 [Representing] フィールドで、[Expression Builder] ボタンをクリックします。[Expression Builder] ウィンドウが表示されます。



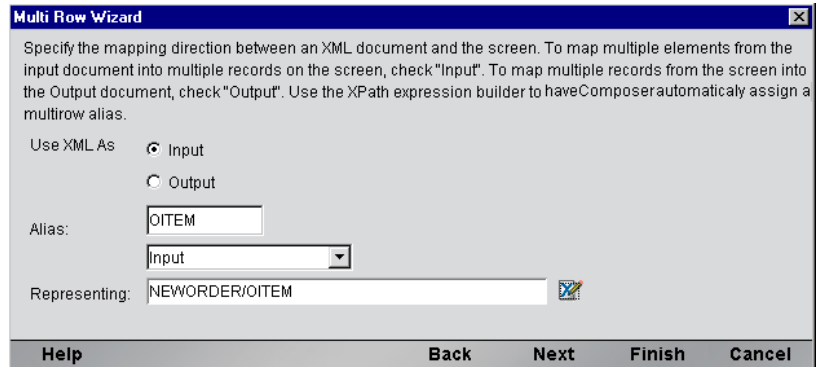
- 4 [XPath Content] フレームで、[Input] 要素を展開します。

- 5 [OITEM] 要素に移動し、ダブルクリックします。ウィンドウの下部にあるコメントペインに式が表示されます。この例では、[OITEM] がループされる要素です。



- 6 [OK]をクリックします。Multi Row ウィザードの2番目の画面に戻ります。ここでは、[Representing] フィールドに新しい式が表示されています。

注記： Expression Builder を使用すると、Composer により自動的に「別名」が作成されます。この例では、Composer で OITEM という別名が作成されています。別名の詳細については、『exteNd Composer ユーザガイド』を参照してください。

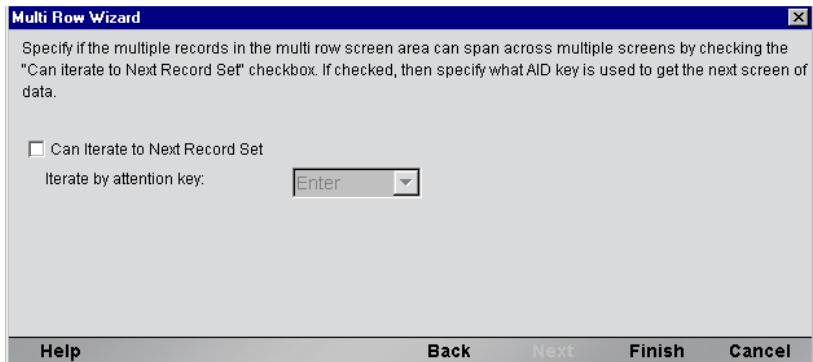


- 7 [Next] をクリックします。Multi Row ウィザードの繰り返しに関する画面が表示されます。

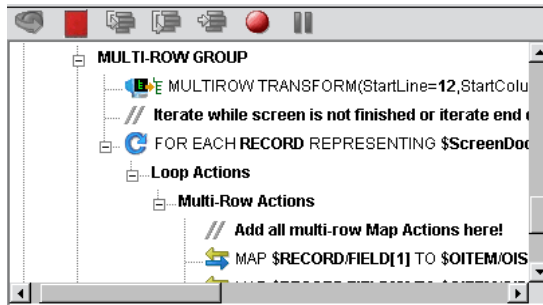
➤ 次のレコードを繰り返す

注記： データを出力する場合、71 ページ「Multi Row ウィザードを使用してデータを出力する」を参照してください。

- 1 この画面では、ページの最後に実行する操作を Multi Row ウィザードに指示します。この例では (60 ページ「Multi Row ウィザードを使用したデータの入力を準備する」を参照)、1 ページ分のデータのみを処理しているため、[Can Iterate to Next Record Set] チェックボックスはオンになっていません。



- 2 [Finish] をクリックします。Multi Row ウィザードで作成したアクションがアクションモデルペインに表示されます。ここでは、[Add all multi row map actions here!] というコメントがハイライトされています。



➤ データの入力に使用する Multi Row アクションを追加する

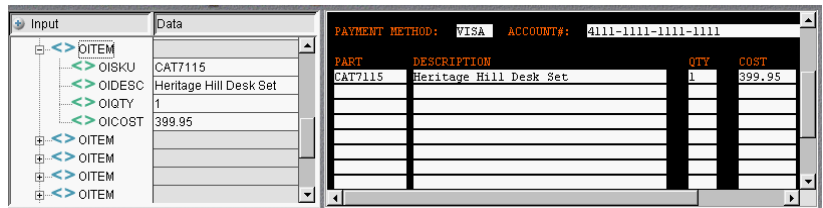
- 1 アクションモデルで、[Add all multi row map actions here!] というコメントがハイライト表示されていない場合は、選択してハイライトします。
- 2 入力 DOM ペインで、OITEM の最初のインスタンスに移動します。

- 3 OITEM のチャイルド要素を、5250 画面の品目のフィールドにある、最初の行の該当するフィールドにドラッグアンドドロップします (次の図を参照)。

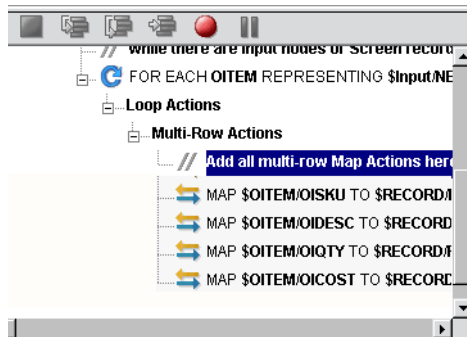
表 4-3

マップ元: 入力 DOM	マップ先: 5250 の [ENTER ORDER 3270] 画面
OISKU	PART
OIDESC	DESCRIPTION
OIQTY	QTY
OICOST	COST

入力 DOM と 5250 の [ENTER ORDER 3270] 画面が次のように表示されます。

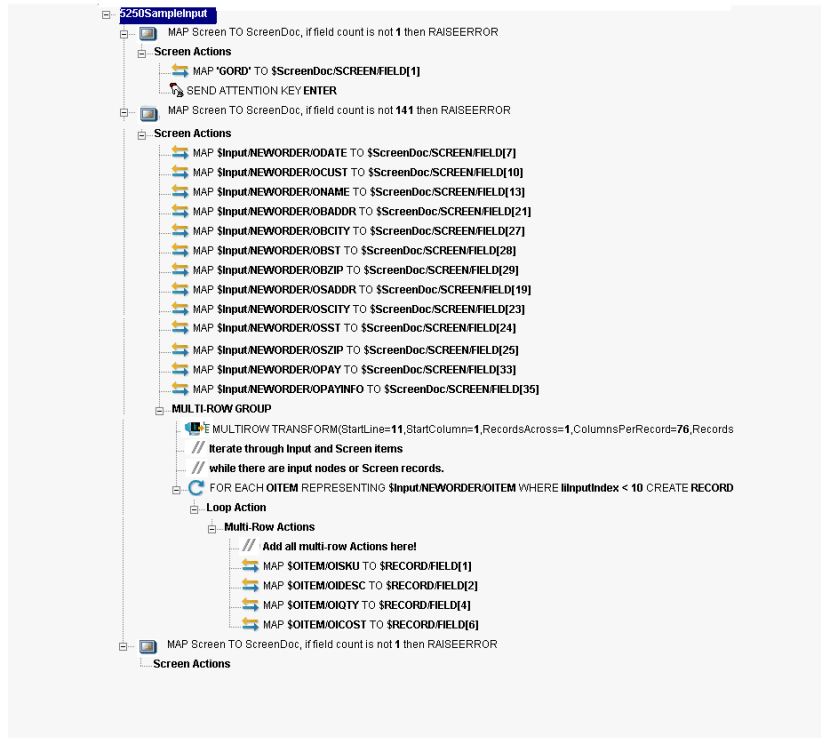


アクションモデルペインが次のように表示されます。



Multi Row アクションのブロック内にある Map アクションは、画面内の絶対的な位置ではなく、行内の相対的な位置で参照されることに注意してください。したがって、最初の Map アクションのターゲットは、\$RECORD/FIELD[1] です。Multi Row のコンテキスト内で、ドラッグアンドドロップによって Map アクションを作成すると、これらのフィールドインデックスが自動的に作成され、割り当てられます。

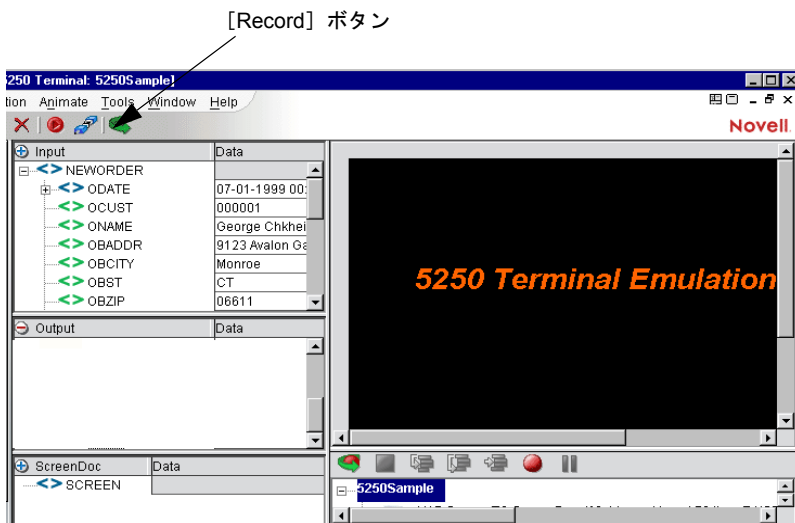
最終的なアクションモデルが次のように表示されます。



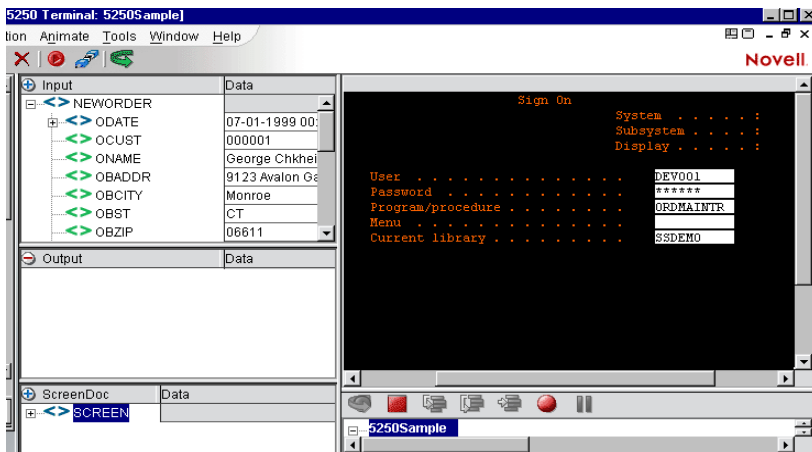
- 4 5250 コンポーネントエディタのメニューバーで、[File]、[Save] の順に選択するか、[Save] ボタンをクリックします。
- 5 81 ページ「アニメーションツールの使用」の指示に従って、コンポーネントをテストします。

➤ Multi Row ウィザードを使用してデータを出力する

- 1 31 ページ「新しい 5250 コンポーネントを作成する」の指示に従って、コンポーネントを作成します。この例では、「5250SampleOutput」というコンポーネントが作成されます。次の図の 5250 コンポーネントエディタに、コンポーネントを示します。

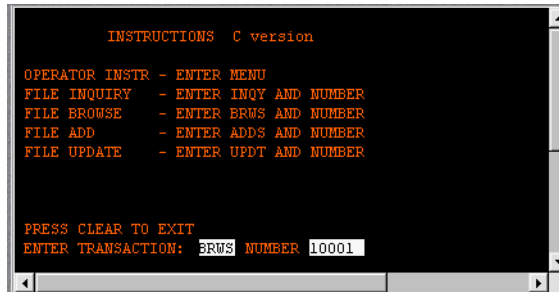


- 2 [Record] ボタンをクリックします。ネイティブ環境ペインにサインオン画面が表示されます。



- 3 この例で使用する適切な画面に移動し、[UserID]、[Password]、および必要な [Program/Procedure] のエントリを入力します。この例では、プログラム / プロシージャとして「MENU」を使用しています。
- 4 キーボードで、<Enter> を押します。この例では、ネイティブ環境ペインに、「INSTRUCTIONS」という 5250 画面が表示されます。

注記： また、ネイティブ環境ペイン内で、5250 のツールバーから、[Enter] を選択することもできます。このツールバーを表示するには、5250 コンポーネントエディタのメニューバーから [View]、[Keypad Toolbar] の順に選択します。

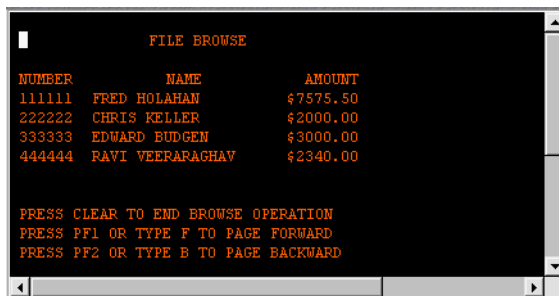


```
INSTRUCTIONS C version

OPERATOR INSTR - ENTER MENU
FILE INQUIRY   - ENTER INQY AMD NUMBER
FILE BROWSE   - ENTER BRWS AMD NUMBER
FILE ADD      - ENTER ADDS AMD NUMBER
FILE UPDATE   - ENTER UPDT AMD NUMBER

PRESS CLEAR TO EXIT
ENTER TRANSACTION: BRWS NUMBER 10001
```

- 5 この例では、[TRANSACTION] フィールドに「BRWS」が、[NUMBER] フィールドに「10001」が、それぞれ入力されています。
- 6 キーボードで、<Enter> を押します。「FILE BROWSE」という 5250 の画面が表示されます。



```
FILE BROWSE

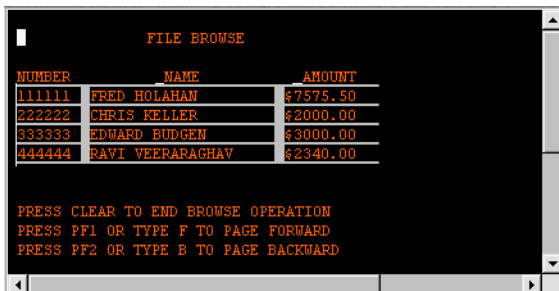
NUMBER      NAME              AMOUNT
111111     FRED HOLAHAN      $7575.50
222222     CHRIS KELLER      $2000.00
333333     EDWARD BUDGEN     $3000.00
444444     RAVI VEERARAGHAV $2340.00

PRESS CLEAR TO END BROWSE OPERATION
PRESS PF1 OR TYPE F TO PAGE FORWARD
PRESS PF2 OR TYPE B TO PAGE BACKWARD
```

- 7 データ領域を特定して、右上隅から左下隅に向かってカーソルをドラッグします。ドラッグマーカーがグレーの背景色でハイライトされます。

注記： カーソルのドラッグは、最初のフィールドの「外側」から開始するようにしてください。フィールド内でドラッグを開始すると、フィールド自体を移動しようとしているとみなされます。

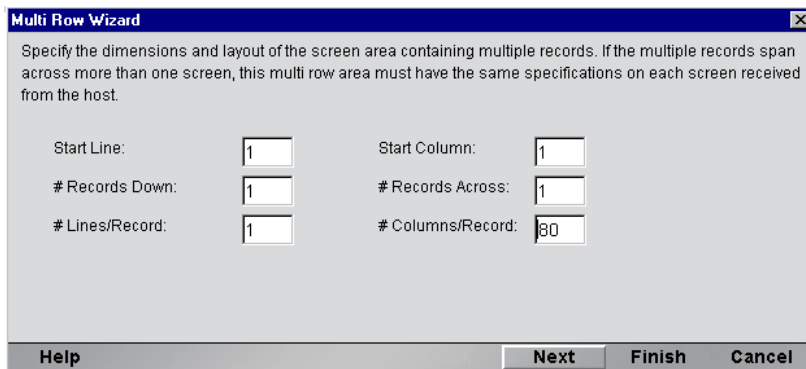
[FILE BROWSE] 画面が次のように表示されます。



NUMBER	NAME	AMOUNT
111111	FRED HOLAHAN	\$7575.50
222222	CHRIS KELLER	\$2000.00
333333	EDWARD RUDGEN	\$3000.00
444444	RAVI VEERARAGHAV	\$2340.00

PRESS CLEAR TO END BROWSE OPERATION
PRESS PF1 OR TYPE F TO PAGE FORWARD
PRESS PF2 OR TYPE B TO PAGE BACKWARD

- 8 5250 コンポーネントエディタから、[Action]、[New Action]、[Multi Row] の順に選択します。Multi Row ウィザードが表示されます。ダイアログボックスには、前の手順でハイライト表示した領域に関するデータが自動的に表示されます。



Multi Row Wizard

Specify the dimensions and layout of the screen area containing multiple records. If the multiple records span across more than one screen, this multi row area must have the same specifications on each screen received from the host.

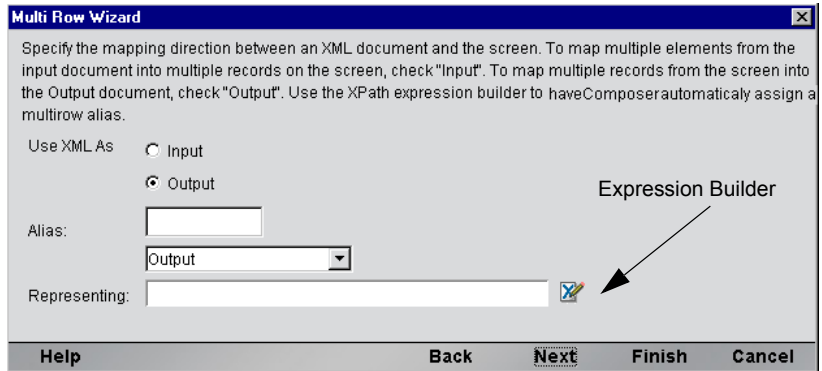
Start Line: Start Column:

Records Down: # Records Across:

Lines/Record: # Columns/Record:

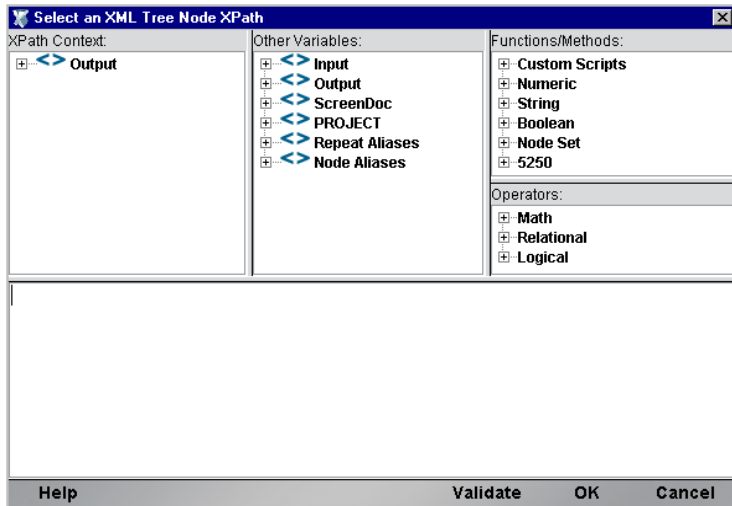
Help Next Finish Cancel

- 9 必要に応じてフィールドを編集します。パラメータが適切である場合、[Next] をクリックします。Multi Row ウィザードの 2 番目のパネルが表示されます。



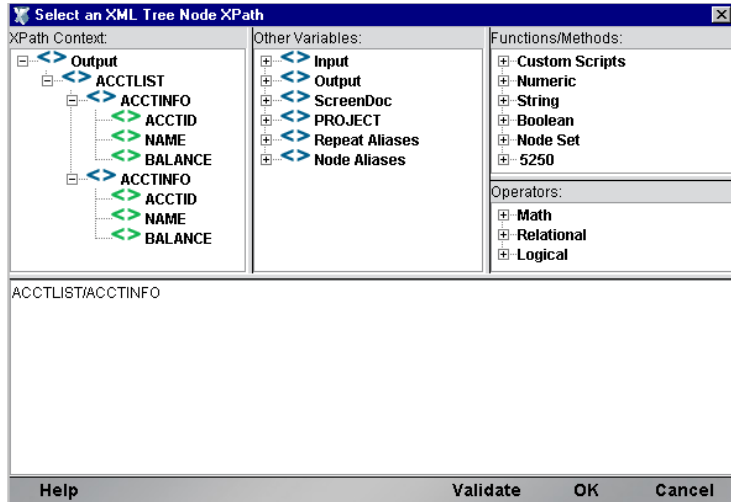
- 10 [Use XML As] で、[Output] ラジオボタンを選択します。このパネルは、複数の要素または画面の行を処理する Repeat アクションを作成するために使用します。これは、すべてのコンポーネントで使用できる基本の「Repeat for Element」アクションと類似したアクションです。

- 11 [Representing] フィールドで、[Expression Builder] ボタンをクリックします。[Expression Builder] ウィンドウが表示されます。



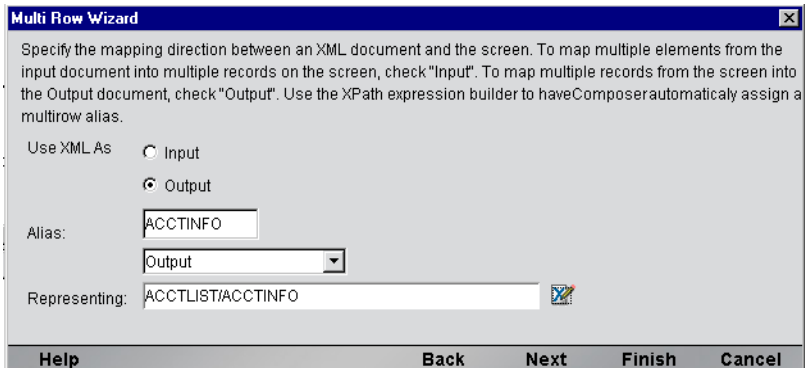
- 12 [XPath Content] フレームで、[Output] 要素を展開します。

- 13 [ACCINFO] 要素に移動し、ダブルクリックします。ウィンドウの下部にあるコメントペインに式が表示されます。この例では、[ACCINFO] がグループされる要素です。

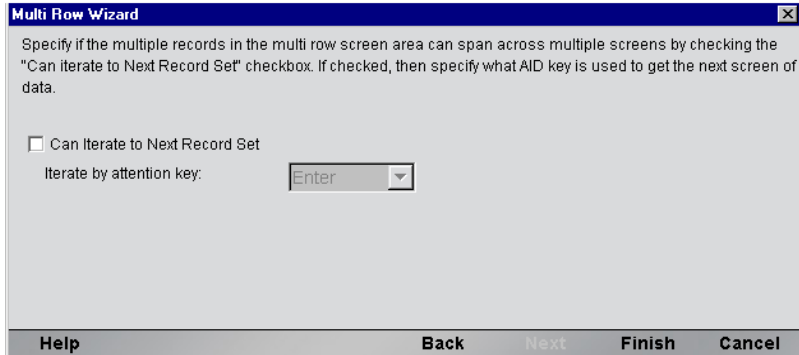


- 14 [OK]をクリックします。Multi Row ウィザードの2番目の画面に戻ります。ここでは、[Representing] フィールドに新しい式が表示されています。

注記： Expression Builder を使用すると、Composer により自動的に「別名」が作成されます。この例では、ACCINFO という別名が作成されています。別名の詳細については、『exteNd Composer ユーザガイド』を参照してください。



- 15 [Next] ボタンをクリックします。Multi Row ウィザードの繰り返しに関する画面が表示されます。この画面では、ページの最後に実行する操作を Multi Row ウィザードに指示します。

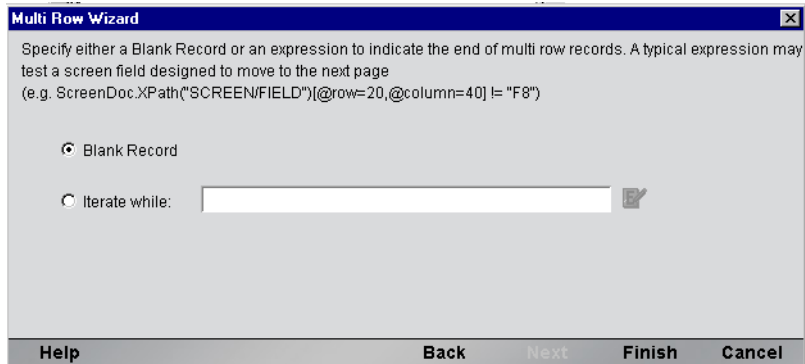


- 16 [Can Iterate to Next Record Set] チェックボックスをオンにします。
- 17 [Iterate by attention key] プルダウンリストで、[F1] を選択します。

注記： 5250 画面には、繰り返しキーに関する説明が表示されます。この例では、[FILE BROWSE] 画面に、<F1> キーを使用して次のページに進むという指示が含まれています。

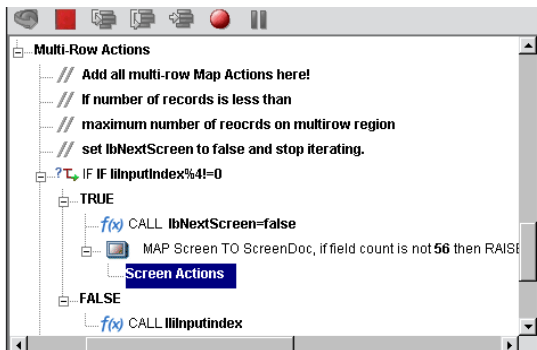
- 18 [Next] をクリックします。

➤ Multi Row ウィザードで、境界のパラメータに関するページを設定する



- 1 [Blank Record] ラジオボタンを選択します。これにより、空白のレコードは Loop アクションのエンドポイントであるとウィザードに指定します。Expression Builder を使用して、Loop アクションのエンドポイントに別のインジケータを設定することもできます。Expression Builder の使用の詳細については、『exteNd Composer ユーザガイド』を参照してください。

- 2 [Finish]をクリックします。Multi Row ウィザードで作成したアクションがアクションモデルペインに表示されます。ここでは、[Add all multi row map actions here!] というコメントがハイライトされています。



➤ データの出力に使用する Multi Row アクションを追加する

- 1 アクションモデルで、[Add all multi row map actions here!] というコメントがハイライト表示されていない場合は、選択してハイライトします。
- 2 出力 DOM ペインで、ACCINFO の最初のインスタンスに移動します。
- 3 5250 の [FILE BROWSE] 画面のデータを出力 DOM にドラッグアンドドロップします (次を参照)。

表 4-4

マップ元: FILE BROWSE	マップ先: 出力 DOM
NUMBER	ACCTID
NAME	NAME
AMOUNT	BALANCE

出力 DOM が次のように表示されます。

The screenshot shows a Novell 5250 Terminal window with the following components:

- Input Data:** A table with columns for PART, DESCRIPTION, QTY, COST, and EXT. The data is as follows:

PART	DESCRIPTION	QTY	COST	EXT
NG1610	16' ANGLE 100 COUNT	1	349.95	349.95
NG1620	16' ANGLE 200 COUNT	2	594.50	1189.00
NG3210	32' ANGLE 100 COUNT	1	594.50	594.50
NG3220	32' ANGLE 200 COUNT	3	175.00	525.00
HN1605	16' CHANNEL 50 COUNT	2	498.99	997.98
HN1610	16' CHANNEL 100 COUNT	2	995.98	1991.96
HN2405	24' CHANNEL 50 COUNT	1	789.99	789.99
HN2410	24' CHANNEL 100 COUNT	2	245.00	490.00
- Output Data:** A table with columns for OITEM, OISKU, OIDESC, OIQTY, and OICOST. The data is as follows:

OITEM	OISKU	OIDESC	OIQTY	OICOST
	ANG161	16' ANGI	1	349.95
- ScreenDoc:** A list of fields for data entry.
- Screen Actions:** A list of actions including:
 - CALL IbmNextScreen=false
 - MAP Screen TO ScreenDoc, if field count is not 56
 - FALSE
 - CALL IbmInputIndex
 - MAP \$RECORD.FIELD[1] TO \$OITEM/OISKU
 - MAP \$RECORD.FIELD[2] TO \$OITEM/OIDESC
 - MAP \$RECORD.FIELD[3] TO \$OITEM/OIQTY
 - MAP \$RECORD.FIELD[4] TO \$OITEM/OICOST
 - SEND ATTENTION KEY ENTER
 - MAP Screen TO ScreenDoc, if field count is not 125 then RAISEERROR

- 4 5250 コンポーネントエディタのメニューバーで、[File]、[Save] の順に選択するか、[Save] ボタンをクリックします。
- 5 81 ページ「アニメーションを実行する」の指示に従って、コンポーネントをテストします。

➤ Multi Row アクションを編集する

- 1 アクションペインで、変更する MultiRow アクションをクリックします。ダイアログボックスが表示されます。

The screenshot shows a dialog box titled "Edit Multi-Row Action" with a close button (X) in the top right corner. It has four tabs: "Record Layout", "XML usage", "iterator parameters", and "Output Row Boundaries". The "Output Row Boundaries" tab is selected. Below the tabs, there is a text instruction: "Specify the dimensions and layout of the screen area containing multiple records. If the multiple records span across more than one screen, this multi row area must have the same specifications on each screen received from the host." Below this text are six input fields arranged in two columns. The left column contains "Start Line:" (12), "# Records Down:" (8), and "# Lines/Record:" (1). The right column contains "Start Column:" (1), "# Records Across:" (1), and "# Columns/Record:" (50). At the bottom of the dialog, there are three buttons: "Help", "OK", and "Cancel".

Start Line:	12	Start Column:	1
# Records Down:	8	# Records Across:	1
# Lines/Record:	1	# Columns/Record:	50

- 2 該当するタブをクリックし、フィールドを編集した後、[OK] ボタンをクリックします。MuliRow ウィザードを使用してデータを入出力する方法については、前の節を参照してください。

5250 コンポーネントの実行

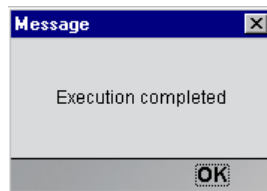
Composer には、コンポーネントをテストできるアニメーションツールが含まれています。5250 コンポーネントエディタのツールバーには、[Execute] ボタンがあり、このボタンを使用するとアクションモデル全体を実行して、コンポーネントが意図したとおりに動作するか検証できます。

➤ 5250 コンポーネントを実行する

- 1 5250 コンポーネントを開きます。5250 コンポーネントエディタウィンドウが表示されます。



- 2 [Execute] ボタンを選択します。アクションモデルのアクションが実行され、アクションの完了時にメッセージが表示されます。



- 3 [OK] をクリックします。
- 4 [View] メニューから、[Expand XML Documents] を選択します。これにより、XML ドキュメントのペアレント、チャイルド、データ要素などがすべて展開され、実行されたコンポーネントの結果を確認できます。XML ドキュメントを展開しないと、指定したデータが、5250 環境から出力 DOM に移動しているかどうかを確認できません。

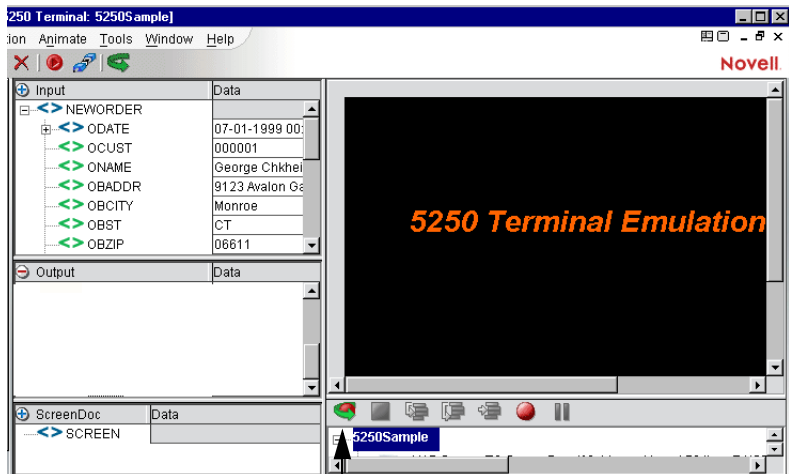
アニメーションツールの使用

アクションモデルには、1 つまたは複数のブレイクポイントを設定することにより、アクションモデルの特定のセクションをテストできるアニメーションツールがあります。これらのツールを使用すると、適切に動作するアクションをすべて実行して、問題の生じたアクションで停止してから、問題のアクションを1つずつトラブルシューティングすることができます。

注記： アニメーションツール機能の簡単な例は、次のとおりです。すべてのアニメーションツールおよびその機能の詳細については、『exteNd Composer ユーザガイド』を参照してください。

➤ アニメーションを実行する

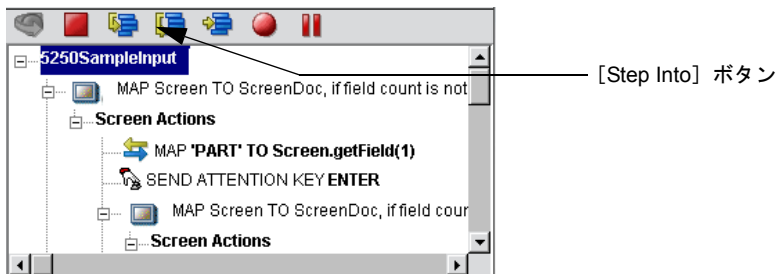
- 1 5250 コンポーネントを開きます。5250 コンポーネントエディタウィンドウにコンポーネントが表示されます。



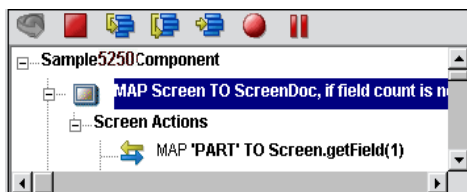
[Start Animation] ボタン

注記： アニメーションモードと記録モードは、コンポーネントでは互いに排他的なモードです。アニメーション中に記録を行うには、アニメーションを一時停止または停止してから、記録モードをオンにする必要があります。

- 2 アクションモデルのツールバーで **[Start Animation]** ボタンをクリックするか、またはキーボードの **<F5>** キーを押します。ツールバーのすべてのツールが有効になります。



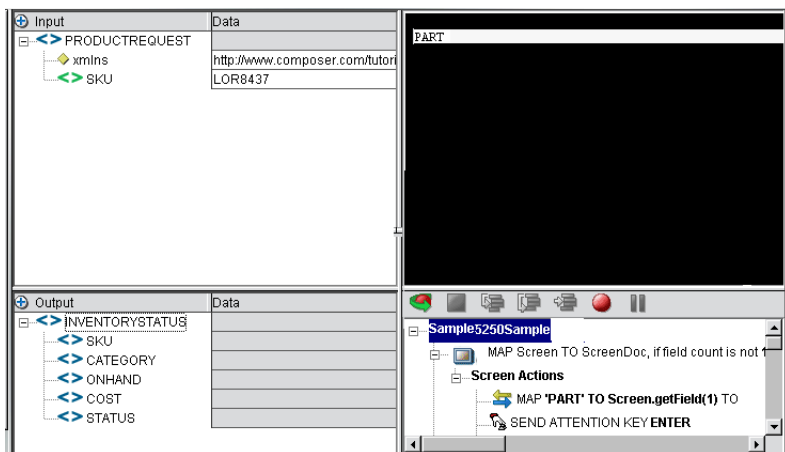
- 3 **[Step Into]** ボタンをクリックします。最初のMapアクションが選択されます。



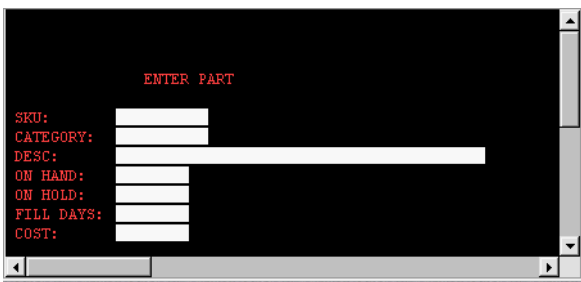
- 4 **[Step Into]** ボタンをもう一度クリックします。ネイティブ環境ペインの入力フィールドに「PART」というコマンドを入力するという指示がハイライト表示されます。



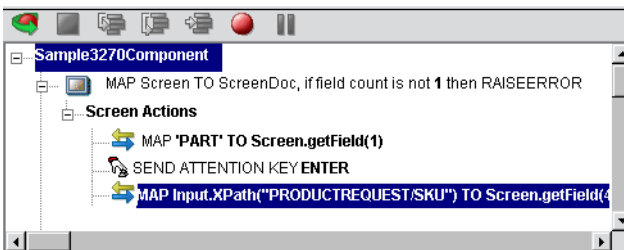
- 5 **[Step Into]** ボタンをもう一度クリックします。ネイティブ環境ペインの入力行に「PART」という語が表示されます。



- 6 [Step Into] ボタンをもう一度クリックします。ネイティブ環境ペインに [ENTER PART] 画面が表示されます。



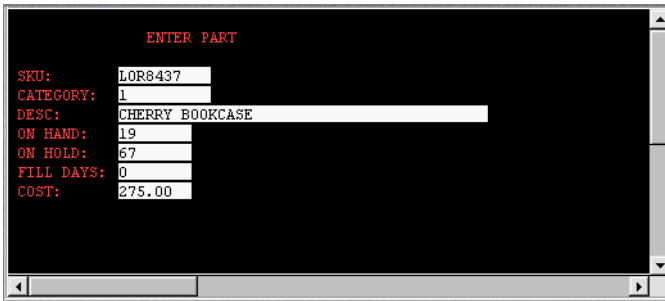
- 7 [Step Into] ボタンをもう一度クリックします。アクションモデルで、SKU を入力 DOM から [ENTER PART] 画面の [SKU] フィールドにドラッグアンドドロップするという指示がハイライト表示されます。



- 8 [Step Into] ボタンをもう一度クリックします。入力 DOM の SKU データが [ENTER PART] 画面の [SKU] フィールドに表示されます。



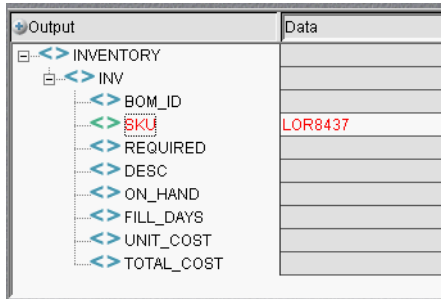
- 9 [Step Into] ボタンをもう一度クリックします。[ENTER PART] 画面に、SKU 番号に関連付けられたデータが表示されます。



- 10 [Step Into] ボタンをもう一度クリックします。アクションモデルで、SKU データを [ENTER PART] 画面から出力 DOM にドラッグするという指示がハイライト表示されます。

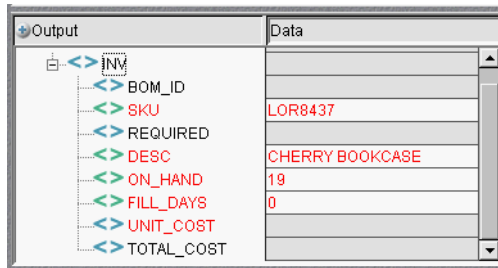


- 11** [Step Into] ボタンをもう一度クリックします。[ENTER PART] 画面の SKU データが出力 DOM の [SKU] フィールドに表示されます。



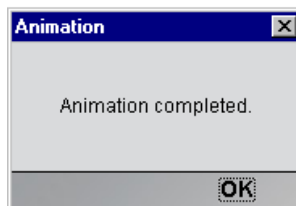
Output	Data
INVENTORY	
INV	
BOM_ID	
SKU	LOR8437
REQUIRED	
DESC	
ON_HAND	
FILL_DAYS	
UNIT_COST	
TOTAL_COST	

- 12** [ENTER PART] 画面のフィールドにあるすべてのデータ要素が出力 DOM に表示されるようになるまで、[Step Into] ボタンをクリックします (次の図を参照)。



Output	Data
INVENTORY	
INV	
BOM_ID	
SKU	LOR8437
REQUIRED	
DESC	CHERRY BOOKCASE
ON_HAND	19
FILL_DAYS	0
UNIT_COST	
TOTAL_COST	

- 13** 完了すると、次のメッセージが表示されます。

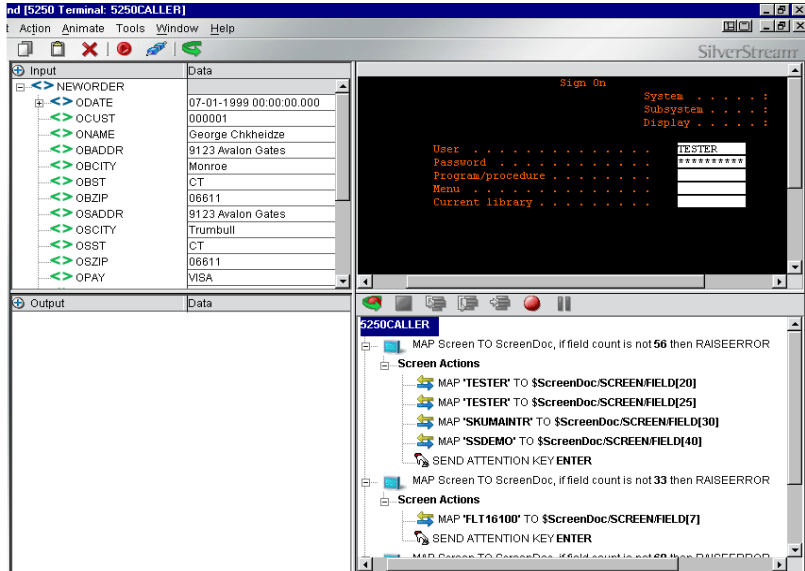


Component with Connection アクション

Component with Connection アクションは、3270 コンポーネントから別のコンポーネントを呼び出し、同じ接続を共有できるため、特殊なアクションであると言えます。このアクションを使用すると、大きなコンポーネントを 3270 のメインコンポーネントとサブコンポーネントに分けることができるため、アクションモデルを簡単に管理できます。メインコンポーネントがサブコンポーネントと接続を共有できると、ランタイム時における接続のオーバーヘッドやトランザクションの移動の量を大幅に減少することができます。作業を開始する前に、必要なサブコンポーネント数を特定し、記録開始時に使用する (アクションが記録されていない) シェルを作成し、保存します。

➤ Component with Connect アクションを使用する

- 1 サブコンポーネントの呼び出しが可能な程度のメインコンポーネントの基本構造を作成および記録します。この例では、メインコンポーネントは「5250 Caller」です。



- 2 メインメニューから、またはマウスの右ボタンをクリックして、[New Action] > [Component/w connection] を選択します。すると、次のダイアログボックスが表示されます。

Component

Component Type: 5250 Terminal Component Name: 5250CALLED

Passed ID	To ID	Template Category	Template Name
Input	Input	{System}	{ANY}

Returned ID	From ID	Template Category	Template Name
Output	Output	{System}	{ANY}

Help OK Cancel

- 3 [Component Type] プルダウンリストから、コンポーネントタイプの名前を選択します。[Component Name] プルダウンリストから、コンポーネントの名前を選択します。
- 4 変更する必要がある場合、渡される ID をプルダウンリストから選択します。変更する必要がある場合、返される ID をプルダウンリストから選択します。[OK] をクリックします。
- 5 アクションがマップペインに表示されます。

[5250 Terminal: 5250CALLER]

File Edit View Options Help

Novell

Input Data

- NEWORDER
 - ODATE: 07-01-1999 00:00:00.000
 - OCUST: 000001
 - ONAME: George Chkheidze
 - OBADDR: 9123 Avalon Gates
 - OBCITY: Monroe
 - OBST: CT
 - OBZIP: 06611
 - OSADDR: 9123 Avalon Gates
 - OSCITY: Trumbull
 - OSST: CT
 - OSZIP: 06611
 - OPAY: VISA

Output Data

SKU Maintenance

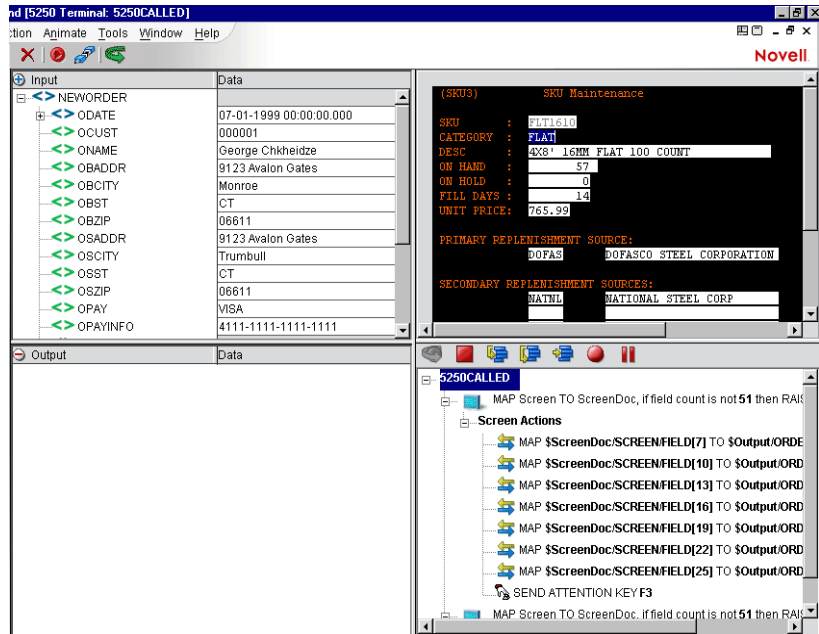
COUNTRY : FL1610
 CATEGORY : FLAT
 AMD : 4X8 16MM FLAT 100 COUNT
 QTY : 50
 OLD : 0
 DAYS : 14
 PRICE : 765.99

PRIMARY REPLENISHMENT SOURCE:
DOFA3 DOFASCO STEEL CORPORATION

SECONDARY REPLENISHMENT SOURCES:
NATHL NATIONAL STEEL CORP

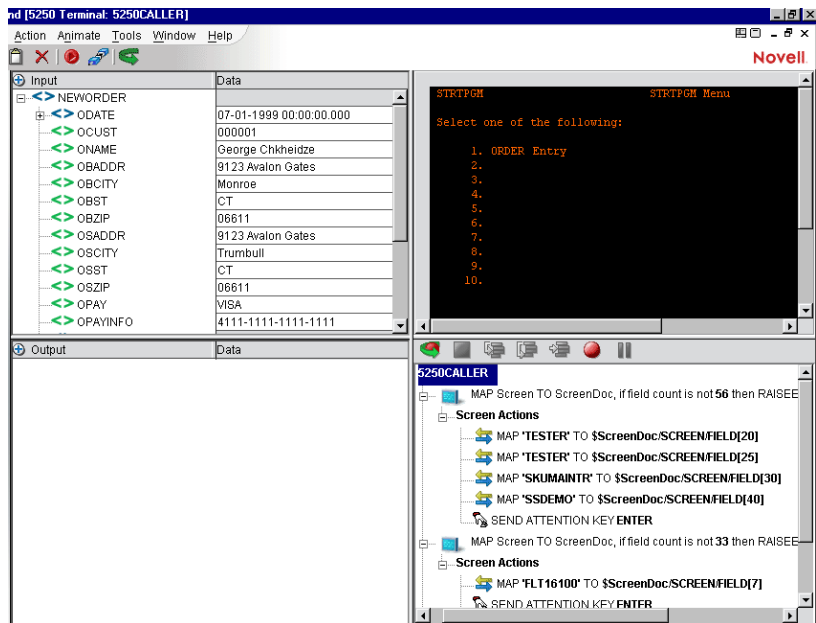
MAP *SKUMAINTR* TO \$\$ScreenDoc\$SCREENFIELD[30]
 MAP *SSDEMO* TO \$\$ScreenDoc\$SCREENFIELD[40]
 SEND ATTENTION KEY ENTER
 MAP Screen TO ScreenDoc, if field count is not 33 then RAISEERROR
 Screen Actions
 MAP *FLT16100* TO \$\$ScreenDoc\$SCREENFIELD[7]
 SEND ATTENTION KEY ENTER
 MAP Screen TO ScreenDoc, if field count is not 68 then RAISEERROR
 Screen Actions
 Execute 5250CALLED Pass(input) Return Output Passing connecto
 MAP Screen TO ScreenDoc, if field count is not 51 then RAISEERROR
 Screen Actions

- 6 Mapコンポーネントをアニメーション表示し、Component with Connection アクションをステップインすると、サブコンポーネントが開きます。画面の表示が「5250Called」というコンポーネントに変更されることを確認します。

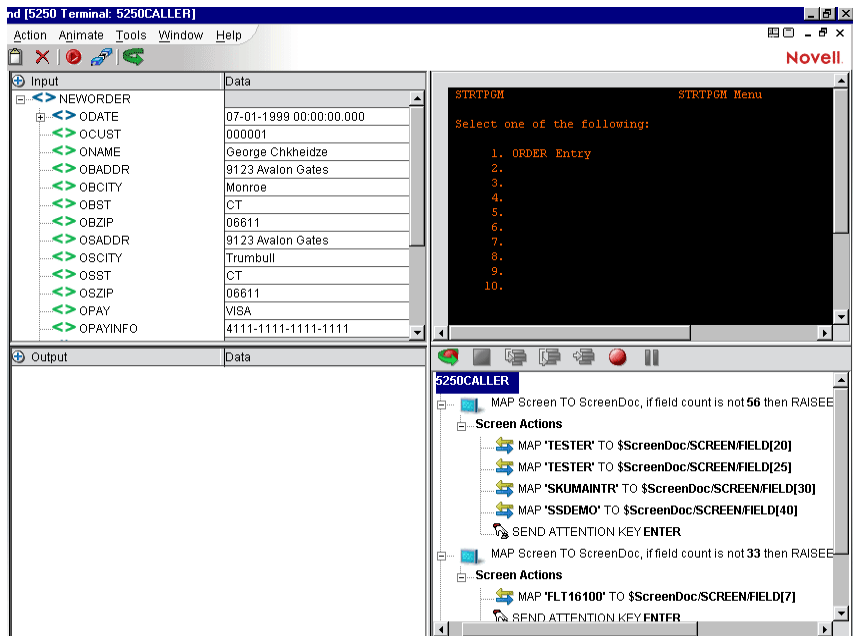


- 7 アニメーションツールバーで [Pause] ボタンをクリックし、[Record] ボタンを有効にします。

- 8 [Record] ボタンをクリックし、サブコンポーネントのアクションを記録します。



- 9 記録処理を停止する前に、コンポーネントを保存します。
- 10 5250 Called のウィンドウが 5250 Caller のウィンドウに戻ります。

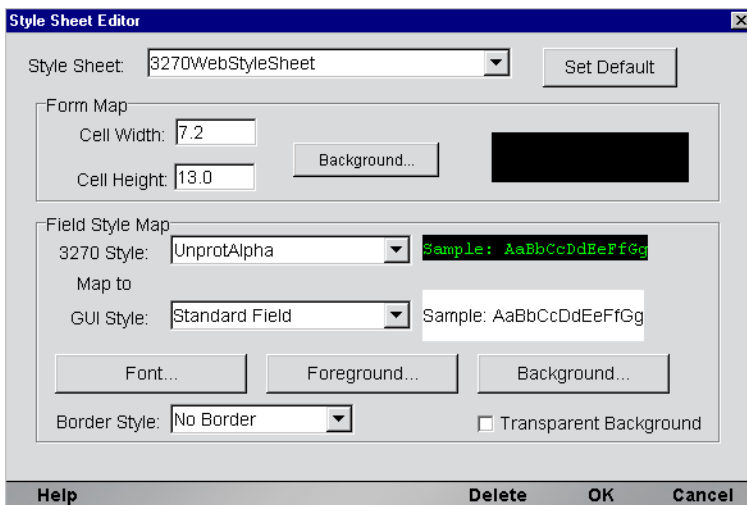


ネイティブ環境ペインでのスタイルシートの使用

5250 コンポーネントエディタのスタイルシートの機能では、ネイティブ環境ペインの表示方法に関するオプションを選択できます。

➤ ネイティブ環境ペインにスタイルシートを適用する

- 1 5250 コンポーネントエディタウィンドウの **[Component]** メニューから、**[Style Sheets]** を選択します。**[Style Sheet Editor]** ウィンドウが表示されます。



- 2 **[Style Sheet]** ドロップダウンリストから **[Style Sheet]** を選択します。Style Sheet Editor の使用の詳細については、27 ページ「スタイルシートリソースを作成する」を参照してください。
- 3 **[OK]** をクリックします。

5250 コンポーネントエディタでの他のアクションの使用

5250 コンポーネントエディタでは、Map Screen アクションおよび Multi Row アクションに加え、その他すべてのアクションを使用できます。[Action] メニューには、基本的なアクションおよび高度なアクションの両方のリストが表示されます (次の表を参照)。

表 4-5

基本的なアクション	説明
Comment	アクションモデルを記録します。特に、アクションモデルに Decision または Repeat、あるいはその両方が使用されている場合、コメントを使用して処理を明確にすることができます。
Component	別のコンポーネントを実行し、呼び出されたコンポーネントで受け渡しするランタイム DOM を指定します。
Decision	指定した条件に基づいて、アクションの 2 つのセットから 1 つを実行できます。コンポーネントの実行で指定した条件がどのように解決されるかによって、True または False へのパスの分岐を処理します。
Function	ECMAScript スクリプト関数または以前に作成したカスタムスクリプトのいずれかを実行します。カスタムスクリプトは、Composer のカスタムスクリプトリソースエディタを使用して作成できます
Log	コンポーネントに指定されているさまざまなログファイルに情報を書き込みます。ログのタイプには、システム出力、システムログ、およびユーザログの 3 種類があります。
Map	要素のデータのある XML DOM から別の XML DOM へ転送し、オプションで変換します。
Send Mail	コンポーネントの実行中、指定した電子メールアドレスに自動的に電子メールを送信します。
Switch	入力値とケースの値との一致に基づいて、プログラムの制御をアクションの特定のブロックに分岐させることができます。これは、長く、読み取りが困難な if/else (Decision アクション) のチェーンを排除するときに使用できる、基本的に便利なアクションです。

表 4-6、4-7、および 4-8 の各アクションは、コンポーネントエディタの [Action] メニューで、[Advanced]、[Data Exchange and Repeat] の順にサブメニューを選択すると利用できます。

表 4-6

高度なアクション	説明
Apply Namespaces	NameSpace プリフィックスを上書きしたり、新しい NameSpace プリフィックスを宣言したり、または NameSpace 全体を無視したりすることができます。
Raise Error	条件を評価し、true となる場合は、ERROR と呼ばれるグローバル変数に式のコンテンツを記述します。単独で使用された場合は、例外をスローしてコンポーネントを停止し、サービスに制御を返します。Try On Error アクションの Execute 分岐内で使用された場合は、評価され、On Error 分岐でアクションに制御が渡されます。
Simultaneous Components	2 つまたはそれ以上のコンポーネントを同時に (つまり、マルチスレッド方式で) 実行できるようにします。
Transaction	非コンテナ管理サービスの一部として配備されるコンポーネントで User Transaction コマンド (開始、コミット、およびロールバックなど) を呼び出したり、コンテナ管理 EJB 配備の一部となるコンポーネントで setRollbackOnly を呼び出したりできます。
Try On Error	一連のアクションを実行することで、エラーを生成するアクションに応答します。Try On Error アクションは、本質的にエラートラップおよび解決を行うアクションです。
XSLT Transform	XSL ファイルの指示に従って XML ファイルを変換します。出力は、一般的に Web ブラウザに XML ファイルを表示するために使用されます。

表 4-7

Data Exchange アクション	説明
UR_/File Read	XML 以外のファイル形式を Composer に読み込むことができます。
UR_/File Write	ファイルを XML 以外の別の形式で書き込むことができます。
WS Interchange	WSDL リソースで定義されたメッセージおよび操作を使用して Web サービスを実行します。
XML Interchange	外部 XML ドキュメントをコンポーネントの DOM に読み込んだり、外部 XML ドキュメントにコンポーネントの DOM を書き込んだりします。読み込み / 書き込みメソッドには、File、FTP、HTTP、および HTTPS プロトコルを使用した Get、Put、Post、および Post with Response メソッドが含まれます。

表 4-8

Repeat アクション	説明
Break	Repeat for Element、Repeat for Group、または Repeat While ループの実行を停止し、ループ外で次のアクションの実行を続行します。
Continue	Repeat for Element、Repeat for Group、または Repeat While ループで現在のループ反復の実行を停止し、次の反復で同じループの一番上から続行します。
Declare Group	複数回発生する要素に基づいてグループを作成し、グループに名前を付けることができます。グループは、Repeat for Group アクションで使用されます。
Repeat for Element	DOM ツリーに指定した要素が発生するごとに 1 つまたは複数のアクションを繰り返します。Repeat for Element アクションでは、複数回発生する要素に基づき、ループを作成できます。
Repeat for Group	グループの各メンバーに対して 1 つまたは複数のアクションを繰り返します。Repeat For Group アクションでは、データを再作成して、データを集約計算できます。
Repeat While	ループを作成することで、1 つまたは複数のアクションを繰り返します。Repeat While アクションでは、処理ループを任意の有効な ECMAScript 式に基づかせることができます。

エラーおよびメッセージの処理

この節では、アニメーションツールの実行時に発生する可能性がある一般的なエラーについて説明します。

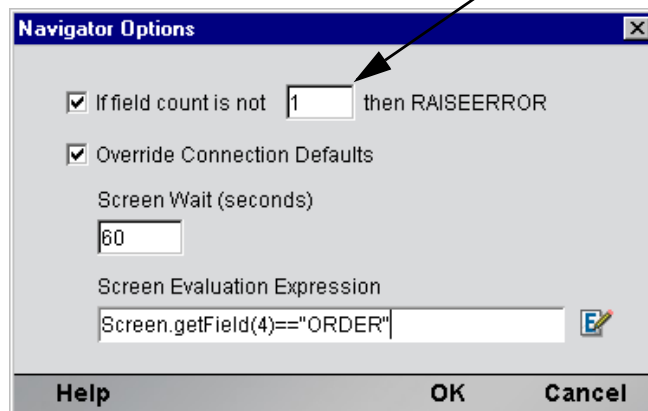
画面フィールド数の変更

このエラーは、特定のトランザクションのアニメーション表示または実行で発生します。このエラーを招く原因の1つに、トランザクションが1つまたは複数の画面を、ユーザからの応答（つまり、Aid キーを押す）を「必要としない」端末に送信した後に、応答を「必要とする」画面を送信する場合があります。たとえば、トランザクションには、メッセージ画面（例：“Please wait...”）を表示した後に、ユーザが実際に使用するトランザクション画面を表示するものがあります。画面表示はトランザクションに制御されるため、ユーザはメッセージ画面に応答できなくなります。トランザクションがこのように動作する問題は、アニメーション表示中に発生します。アクションからアクションへ移行する際、Composer の Map Screen アクションにより、各画面のフィールド数がカウントされ、コンポーネントに記録されている元のフィールド数と比較されます。トランザクションは、対応する Map Screen アクションに移行する前に 2 番目の画面を送信できるため、フィールド数の同期が取れなくなります。

このエラーを修正するには、フィールド数が変更された理由を特定し、次の修正アクションのうちの1つを実行します。

- ◆ アクションモデルで、失敗した Map Screen アクションをダブルクリックし、フィールド数の変数を正しいフィールド数に変更します。
- ◆ Map Screen アクションをダブルクリックし、フィールド数のチェックを無効にすると（次の図を参照）、Map Screen アクション全体を Try/OnError アクション内で折り返します。Try/OnError アクション内では、例外を返すのではなく、フィールド数のエラーを条件に従って処理できます。

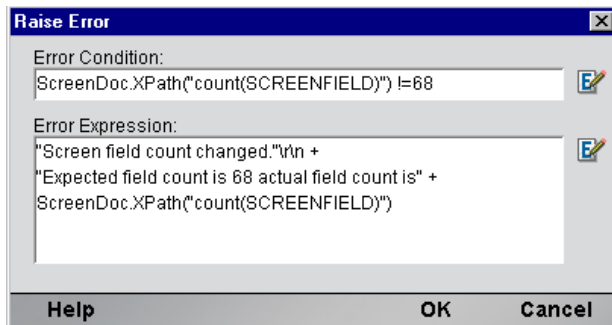
Map Screen フィールドの数をチェックするボックス



[Navigator Options] ダイアログボックスでは、チェックする「フィールドカウント」（フィールド数）を指定できます。ランタイム時にフィールドカウントが適切でない場合、エラーが発生します。（前の手順で適用した）Try/OnError アクションにより、このエラーをトラップします。

注記: 配備する前に、チェックボックスを再度オンにすることを忘れないでください。

- [override connection defaults] をオンにすると、ナビゲーションアクションの [Screen Wait] フィールドに入力した秒数に従って上書きできます。
- パケットが受信され、処理されるたびにチェックされる式を設定します。
- 画面の評価式が true の場合、次の手順 (画面フィールドのチェックを含む場合がある) に進みます。
- Map Screen フィールドの数をチェックするボックスを無効にし、フィールド数をチェックする独自のアクションを追加します。これには、アプリケーションに必要なフィールド数をチェックする Raise Error アクション (例: ScreenDoc.XPath("count(SCREEN/FIELD)")>68) を作成する方法もあります。"例は次のとおりです。



フィールド数の同期が取れなくなる場合、次のような原因も考えられます。

- ◆ プリントジョブから予期しないスプーラメッセージを受信した場合。これを修正するには、アクションモデルで、不適切な Map Screen アクションを削除します。
- ◆ アプリケーションが同じ画面を異なるフィールド数で送信した場合。

接続に関するエラー

接続プールを使用し、不正なユーザ ID またはパスワードを使用してログオンしようとした場合、接続インスタンスは使用できなくなり、プールのメンバーは次の接続要求でスキップされます。「Logon connection in pool <Pool name> was discarded for User ID <User ID>」というエラーメッセージがサーバログに送信されます。運用前のテスト中またはパフォーマンスの問題が生じた場合、あるいはその両方で、この種のメッセージを確認する必要があります。

不正な ID またはパスワードを解決するには、いくつかの異なる手順を使用できます。

- ◆ IDとパスワードが機能しないことをシステム管理者に伝えます。システム管理者が確認した結果、IDとパスワードに問題がない場合、Connection Managerコンソールでサーバをリセットします。
- ◆ IDとパスワードに問題がある場合、接続プールから削除します。

5

ログオンコンポーネント、接続、および接続プール

5250 端末セッションパフォーマンスについて

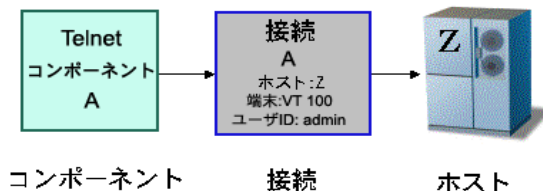
通常の5250端末コンポーネントは、exteNd Composer内のテストワークステーションで適切に動作しますが、運用アプリケーションサーバ環境に配備すると、このコンポーネントをカプセル化するサービスの動作が負荷により遅くなる場合があります。これは、珍しいことではなく、複数のユーザをサポートするデータベースシステムで過去によく見られた問題に類似しています。この問題の原因は、一般的な端末セッショントランザクションで必要となるさまざまな手順に要した時間に通常は由来しています。トランザクション自体の実際の実行を除くと、この手順は次のようになります。

- 1 ホストへの接続を保護する
- 2 ユーザを認証する
- 3 システム内を移動してトランザクションを起動できる場所をポイントする
- 4 トランザクション終了時に、ユーザをサインオフし、接続を切断する

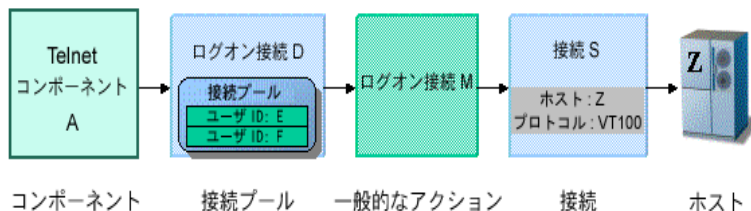
個々のトランザクションに対する負荷は小さい場合でも、多数のWebサイトやアプリケーションサーバ環境で一般的に見られるように、トランザクションの負荷が重くなると、5250端末コンポーネントとセッションのオーバーヘッドの1対1の関係に問題が生じる場合があります。exteNd Composerでは、2つの特別なオブジェクトを用意することで、繰り返しセッションのオーバーヘッドを最小限に抑えています。これらのオブジェクトは、5250ログオン接続と呼ばれる特別な接続リソースタイプと、5250ログオンコンポーネントです。これらのオブジェクトによって、多くの要求が使用する接続と画面の位置を確立、維持できるため、ユーザはAS/400への接続を自由に拡張できます。

接続プールアーキテクチャ

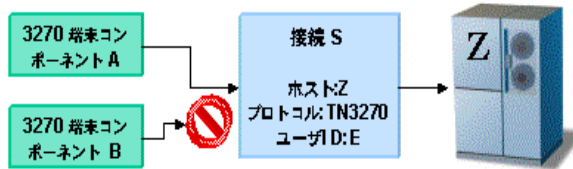
5250 Terminal Connect をインストールすると、TN5250 接続、5250 ログオン接続 (これ以降、ログオン接続と呼びます) という 2 種類の接続リソースが接続の作成ウィザードに追加されます。TN5250 接続は、実際の接続で、5250 端末コンポーネントによって使用されるとホストシステムとのセッションを確立できます。



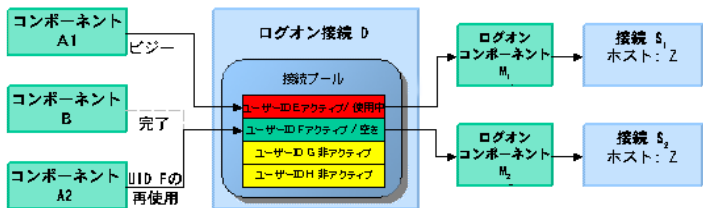
逆に、ログオン接続は異なり、使用可能なユーザ ID のプールを定義し、5250 ログオンコンポーネント (これ以降、ログオンコンポーネントと呼びます) を使用し、個々のユーザ ID に対して接続とナビゲーションに関連するアクションを実行します。実際には、ログオンコンポーネントにより、TN5250 を使用して接続が確立されます。ログオンコンポーネントについては、後に説明しますが、接続プールを確立するには、ログオン接続とログオンコンポーネントを同時に使用する必要があることに注意してください。



通常、5250 端末コンポーネントにより、単一のユーザ ID とパスワードを使用して定義した接続がアクティブになると、その接続のユーザ ID は、同じ接続定義を使用するコンポーネントの別のインスタンス、または別のコンポーネントでは使用できなくなります。



ログオン接続では、追加のユーザ ID を作成して新しい接続を確立することで、他のコンポーネントが終了するまでのシリアル待機時間を減らし、さらに可能な場合は接続を再使用してセッションのオーバーヘッドを回避することで、パフォーマンス上の利点を提供されます。

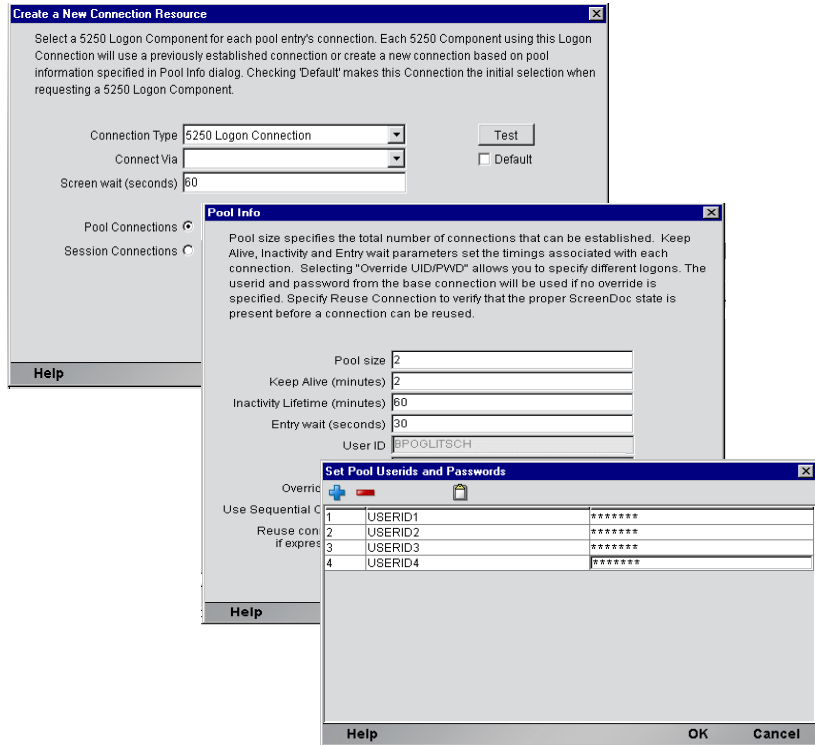


前の図では、アクティブな 5250 端末コンポーネントには、それぞれ独自のユーザ ID、ログオンコンポーネントの独自のインスタンス、および接続リソースの独自のインスタンスがあります。また、同じ接続リソースを使用し、シリアルログオンのオーバーヘッドを繰り返し引き起こす、複数の 5250 端末コンポーネントを実行すると、単一のログオン接続により提供されたユーザ ID / ログオンコンポーネント / 接続インスタンスを代わりに再使用できます。さらに、設計時には、ユーザは 1 つのログオン接続オブジェクト、1 つのログオンコンポーネントオブジェクト、および 1 つの接続オブジェクトのみを作成しています。ログオン接続では、ランタイム時に各ユーザ ID に対して個々のインスタンスを作成します。

ログオン接続、ログオンコンポーネント、および接続の組み合わせは、接続プール要素を構成するものです。追加の接続プールを定義して配備する必要があるかどうかを決定する際の重要な要素として、1 つまたは複数の 5250 端末コンポーネントでログオンコンポーネントの異なる起動画面を使用する必要が生じた場合が挙げられます。接続プールを使用して効率性とパフォーマンスを向上させるために重要なのは、5250 端末コンポーネント、ログオン接続、およびログオンコンポーネントで起動画面が適切に管理されていることを確認することです。

5250 ログオン接続について

ログオン接続は、TN5250 接続リソースのような実際の接続オブジェクトではなく、ユーザ ID に関連付けられたさまざまな接続管理パラメータを持つユーザ ID のプールです。重要なパラメータの 1 つは、初期ログオンタスクと起動画面へのメニューナビゲーションを行うユーザ ID すべてに対して同じログオンコンポーネントの異なるインスタンスを使用することです。



ログオン接続では、ログオンコンポーネントの指定に加えて、次のユーザ ID プール機能が提供されます。

- 1 ユーザ ID が必要となった場合に、クライアントが安全に接続できるよう事前に複数のユーザ ID の仕様を許可する。
- 2 ユーザ ID/ 接続が確立されたら、ユーザ ID/ 接続の再使用を許可してユーザ認証と接続の切断が繰り返し行われないようにする。
- 3 ホストシステムによりサポートされている場合、単一のユーザ ID で複数の接続の使用を許可する。
- 4 接続がアクティブでない間にホストがタイムアウトにならないよう、接続をアクティブ状態に保つ。

- 5 アクティブなプールから接続を削除する時間を指定する。
- 6 完全にアクティブなプールによって空き接続が提供されるまで待機するタイムアウト時間を設定する。
- 7 ログオン接続で使用するログオンコンポーネントの状態に依存するエラーの処理を指定する。

5250 端末コンポーネントからの複数のインスタンス、または異なる 5250 端末コンポーネントで、同じログオン接続を使用するには、次の条件を満たす必要があります。

- 1 すべての5250端末コンポーネントでは、同じ接続リソースを使用する(その結果、TN5250 ホスト、ポート、データエンコードパラメータ、または EPI 5250 接続ゲートウェイ / サーバパラメータを共有する)必要があります。
- 2 すべての 5250 端末コンポーネントには、ホストシステム内にコンポーネントの実行開始場所となる共通の起動画面が必要です(詳細については、「5250 ログオンコンポーネントについて」を参照してください)。

シングルサインオンを使用した接続プール

ホストシステムのセキュリティで、単一のユーザ ID からの複数のログインをサポートしている場合、この単一のユーザ ID をプールするような状況が生じる可能性があります。これを実現するには、次の手順に従います。

- ◆ ログオンコンポーネントで使用する接続リソースでユーザ ID/ パスワードを指定します。
- ◆ ログオン接続の [Pool Info] ダイアログボックスで、[Pool Size] に 1 より大きい値を指定します。
- ◆ ログオン接続の [Pool Info] ダイアログボックスで、[Override the UID/PWD] 設定をオフのままにします。

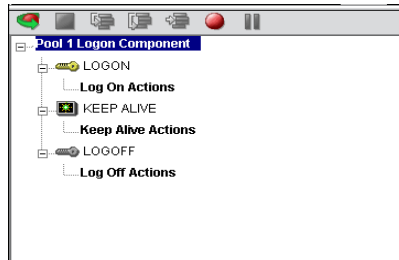
この手順を実行すると、各プールスロットで、接続オブジェクトに含まれるユーザ ID とパスワードが使用され、プールからのユーザ ID は使用されません。

5250 ログオンコンポーネントについて

ログオンコンポーネントは、複数の 5250 端末コンポーネントで使用する接続を管理できるようにアクションモデルが設計されている特別なコンポーネントです。ログオンコンポーネントは、2つの主な違いを除き、5250 端末コンポーネントとほぼ同じです。

- 1 ログオンコンポーネントのアクションモデルは、Logon アクション、KeepAlive アクション、および Logoff アクションから構成される接続タスクにより編成、実行されます。

- 2 ログオンコンポーネントは、別のコンポーネントやサービスによっては実行されず、ログオン接続によって実行されます。
- 3 ログオンコンポーネントは、ログオン接続と組み合わせて使用する必要があります、また組み合わせてのみ使用できます。



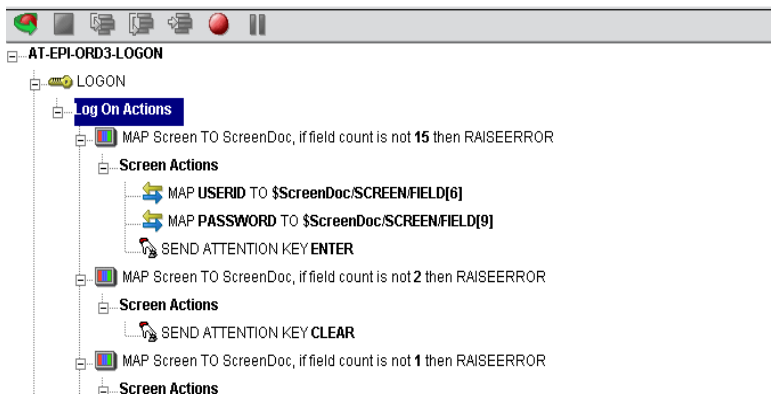
ログオンコンポーネントの接続タスクをログオン接続と組み合わせて使用すると、パフォーマンスにおいてさらに3つの利点があります。

- LOGON アクションは、ホスト環境内を移動し、ホストシステムの希望の「起動画面」で停止します。接続は、プールからの特定のユーザ ID を使用してアクティブにされます。その後、接続を再使用する 5250 端末コンポーネントでは、コンポーネントが起動画面にすでに存在し、新しいセッションを使用した場合と同様に起動画面に移動する際のオーバーヘッドが発生しないため、パフォーマンス上の利点が得られます。
- KEEPALIVE アクションは、ホストが標準のタイムアウト時間内に使用されない場合に接続を切断しないようにし、Logon アクションと同じ起動画面で接続された状態にします。
- LOGOFF アクションは、接続の終了時に、プールからのユーザ ID によって確立された接続すべてに対して、指定した方法でホスト環境を終了します。

注記： LOGON、KEEPALIVE、または LOGOFF アクションの最後のアクションは、アニメーション中には実行されなくても、ログオンコンポーネントがログオン接続コンポーネントとして使用される場合には実行される空の「有効な」ナビゲーションアクションである必要があります。この最後のアクションをユーザが無効にした場合、アニメーションは、実行される2つの連続した空のナビゲーションアクションのために正しく機能しません。たとえば、LOGON のアクションと KEEPALIVE の最初のアクションが無効な場合は、エラーが発生します。

LOGON グループに配置するアクションは、主にホストのセキュリティ画面にサインインしてから、ホストのメニューシステム内を通して起動画面に移動することに関連しています。この起動画面では、各 5250 端末コンポーネントのアクションモデルが開始します。ログオンコンポーネントを使用する 5250 端末コンポーネントでは、共通の画面で実行を開始できなければなりません。それ以外の場合は、移動のオーバーヘッドを回避することでパフォーマンスは向上されず、さらに重要な点として、不適切な 5250 端末コンポーネントが動作しなくなります。

Logon アクションは、ログオン接続を使用しない 5250 端末コンポーネントと同様の方法で作成されます。記録機能を使用し必要なアクションを作成して、ユーザ ID やパスワードなどのサインオン情報と、起動画面に表示する初期メニューの選択項目を入力します。その他の留意点として、ログオン接続プールからユーザ ID とパスワードを使用することが重要です。そのためには、USERID と PASSWORD と呼ばれる 2 つの特別なシステム変数を、画面上の適切なフィールドにマップする必要があります。この 2 つの変数を使用して、exteNd Composer では次のアクティブな空きプールのスロットから値を自動的にマップします。



起動画面は、ログオン接続で提供されたユーザ ID プールを使用した 5250 端末コンポーネントすべてに対する共通の実行ポイントとなる必要があります。起動画面に達するには、通常の 5250 端末コンポーネントでアクションを作成する場合と同じ方法でアクションを作成します。ログオンコンポーネントでは、LOGON アクションは、新しい接続が確立された際に一度だけ実行されます。

ログオンコンポーネントを使用したパフォーマンスの最大化

Logon アクションは、適切な構造でなければならず、前の画面に示すように常に Map Screen アクションで開始および終了する必要があります。最後の Map Screen アクションでは、画面が接続に達する前に 5250 コンポーネントに制御が移らないよう保証されます。それ以外の場合は、5250 コンポーネントが無効な画面で開始して例外がスローされ、おそらく操作が無効になります。Map Screen では、アクションを含める必要はありませんが、5250 コンポーネントから制御が途中で渡されるのを防ぐことは重要です。また、最初の記録セッション中にキャプチャしたものと同一画面であるかどうかを確認するために、画面が到着するとフィールドカウントチェックを実行します。

注記：最後の Map Screen をスキップするログオンコンポーネントをアニメーション表示している場合、これは通常の動作です。運用環境におけるログオンコンポーネントのアクションは、5250 端末コンポーネントでは常にインターリーブ方法で実行されます。最初から最後までログオンコンポーネントをアニメーション表示すると、イベントの異常なシーケンスが実際には作成されます。この結果、2 つの Map Screen が連続して処理されますが、このことは許可されていません。

```
Sign On
System . . . . . : S102P6LM
Subsystem . . . . : QBASE
Display . . . . . : QPADEV001L

User . . . . . _____
Password . . . . . _____
Program/procedure . . . . . _____
Menu . . . . . _____
Current library . . . . . _____

MAIN                               AS/400 Main Menu                               System: S102P6LM
Select one of the following:
1. User tasks
2. Office tasks
4. Files, libraries, and folders
6. Communications
8. Problem handling
9. Display a menu
10. Information Assistant options
11. Client Access/400 tasks
90. Sign off

Selection or command
====>

F3=Exit  F4=Prompt
F23=Set initial menu
(C) COPYRIGHT IBM CO
PAD MSG 1:

STRTPGM                               STRTPGM Menu
Select one of the following:
1. ORDER Entry
2.
3.
4.
5.
6.
7.
8.
9.
10.

Selection or command
====>

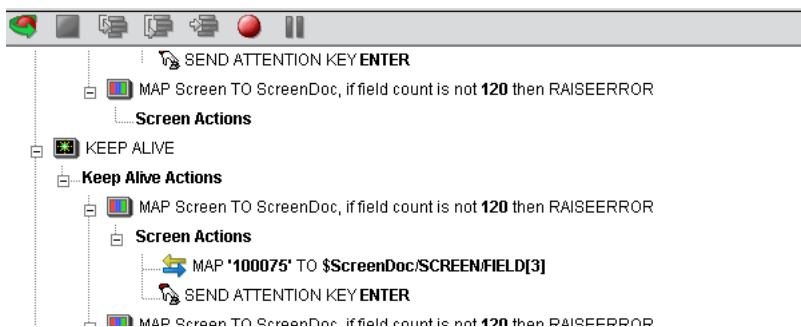
F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel
F13=Information Assistant  F16=AS/400 main menu
PAD MSG 1:AS400 20,07 00:00:251 14:19 1/29/01
```

3つのエントリのユーザ ID プールが完全に使用され、コンポーネントの実行時に15回再使用された場合、目的のトランザクションを実行するメニュー項目に移動する際のオーバーヘッドは、3回だけ発生することになります。同様に、新しい接続がアクティブになった場合（再使用された場合ではなく）には、LOGON アクションが1回だけ実行されるため、ホストへのログオンは3回だけになります。

注記： 可能な場合は、Try/On Error アクションを使用して、回復可能であると思われる潜在的なログオンエラーをトラップします。それ以外の場合は、失敗したログオンを確立しようとするユーザ ID がプールから破棄されるため、5250 の Composer Enterprise Server Console で破棄された接続を手動でリセットするまで、プールサイズは小さくなる可能性があります。詳細については、この章の「プールの管理」を参照してください。

KEEPALIVE アクション

KEEPALIVE 見出しは、アクティビティを作成し、ログオンコンポーネントで使用する接続で発生し続けるホストと通信するアクションを配置する場所です。KEEPALIVE アクションでは、<ENTER> キーなどの AID キーのホストへの送信が通常は行われます。ただし、AID キーの送信後に画面が起動画面とは異なる画面に変わった場合、KEEPALIVE セクションでログオンコンポーネントを起動画面に戻す必要があります。起動画面に戻さなかった場合は、次のコンポーネントが不正な画面で表示され、エラーが発生します。



ログオン接続の [Pool Info] ダイアログボックスでは、KEEPALIVE アクションを実行する頻度を制御します。ログオン接続プールで、空き接続を2分間アクティブに維持するが、アクティビティから1分後にホストにより接続が切断されるように指定した場合、キーボードアクションを指定して（例：<ENTER> キーを送信する）、接続がアクティブであることをホストに通知できます。

Pool size specifies the total number of connections that can be established. Keep Alive, Inactivity and Entry wait parameters set the timings associated with each connection. Selecting "Override UID/PWD" allows you to specify different logons. The userid and password from the base connection will be used if no override is specified. Specify Reuse Connection to verify that the proper ScreenDoc state is present before a connection can be reused.

Pool size: 2

Keep Alive (minutes): 2

Inactivity Lifetime (minutes): 60

Entry wait (seconds): 30

User ID: SCARTER

Password: *****

Override UID/PWD: Set usersids...

Pool Device Names: Set Device Names...

Use Sequential Connections:

Retry discarded after (minutes):

Max retry discarded:

Reuse connection only if expression is true:

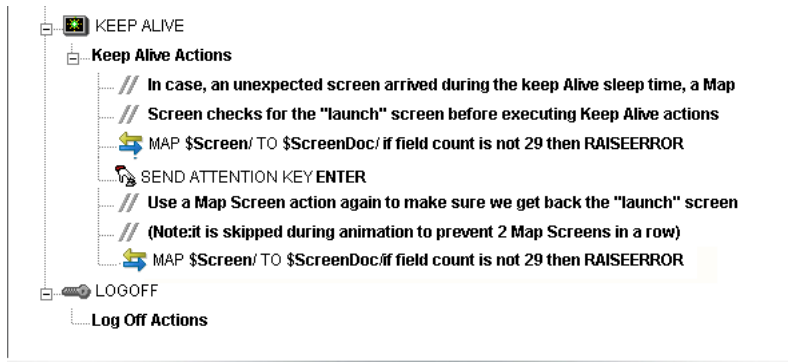
Help OK Cancel

KEEPALIVE アクションは、ログオン接続の [Pool Info] ダイアログボックスで定義した KeepAlive の期間後に、複数回実行できます。

注記: KeepAlive アクションを実行しても、ログオン接続で非活動ライフタイムクロックはリセットされません。非活動ライフタイムクロックがリセットされるのは、5250 端末コンポーネントを実行した場合のみです。

KeepAlive アクションを使用したパフォーマンスの最大化

Map Screen アクションは、Keep Alive セクションの開始時と終了時にも処理されなければなりません。Keep Alive セクションでは、接続が切断されないようにするだけでなく、実行完了時に起動画面が存在するようにする必要があります。最初の Map Screen では、接続が使用可能でも使用されていない間に、ホストから予期しない画面が届いていないことが確認されます。そして再び最後の Map Screen では、Keep Alive アクションの実行後、途中で次の 5250 コンポーネントに接続が解放されないようにします。次の画面を参照してください。



LOGOFF アクション

基本的に、Logoff アクションは、ユーザ ID を適切にホストシステム外に移動させます。Logoff アクションは、接続がタイムアウトになった場合（つまり、非活動ライフタイムの期限が切れた場合）、または接続が 5250 サーバコンソールを介して切断された場合に、1 つの接続に対して一度だけ実行されます。

LOGOFF アクションのパフォーマンスの最大化

Logoff アクションは、安定していなければなりません。Logoff アクションの実行中に例外が発生した場合、exteNd Composer ではホストとの接続が切断され、ユーザ ID がプールで解放されます。ただし、ユーザ ID はホストでアクティブのままになる可能性があります。停止状態が続いたためにホストでユーザ ID が停止されるまで、プールではそのユーザ ID に今後ログオンしようとしても、ログオンをコード化してこの状況に対処しない限り、失敗する可能性があります。ログオンに失敗すると、プールからユーザ ID が破棄され、プールサイズと全体的なパフォーマンスが低減する可能性があります。Logon アクションと KeepAlive アクションに関しては、ログオフの終了時に適切な画面となるよう保証する方法は、Map Screen で終了することです。

ログオンコンポーネントの実行

ユーザ ID がログオン接続プールからアクティブにされるたびに、ログオンコンポーネントのインスタンスが作成され、そのユーザ ID に関連付けられます。その後、目的の起動画面に達するまで、Logon アクションが実行されます。この時点で、5250 端末コンポーネントの実行が開始します。実行が終了すると、同じログオン接続を使用する別の5250端末コンポーネントの実行が同じ起動画面から開始します。

別のコンポーネントの実行が開始しない場合、接続は、ログオン接続の [Pool Info] ダイアログボックスの [Inactivity Lifetime] と [KeepAlive] の設定で定義したアクティブでも空き状態になります。KeepAlive の期間 (例: 2 分) が非活動ライフタイム (例: 120 分) より短い場合は、ホストがタイムアウトして接続が切断されないように、KeepAlive アクションが KeepAlive 期間の終了時に実行され、KeepAlive 期間が最初から開始します。非活動期間と KeepAlive 期間は、ログオン接続の [Pool Info] ダイアログボックスで定義されます。

ログオンコンポーネントの実行ライフタイムは、ログオンコンポーネントを使用するログオン接続のアクティビティに応じて異なります。ログオン接続プールで 1 つのエントリがアクティブの場合は、ログオンコンポーネントの 1 つのインスタンスが、ライブな状態でメモリ内に存在します。非アクティブな状態が続いたために、残っている最後のプールエントリの期限が切れた場合、ログオンコンポーネントの実行は中止されます。ログオンコンポーネントの実行を停止する他の唯一の方法は、サーバの 5250 コンソールを使用することです。

接続プールの作成

概要

5250 端末コンポーネントを作成する場合は、最初に必要な接続オブジェクトを作成しなければなりません。同様に、接続プールを構成するオブジェクトを作成する場合は、最初に必要なオブジェクトを作成しなければなりません。これは、ホストで開始し、ホストにアクセスする 5250 端末コンポーネントに戻ることを意味します。接続プールを作成する標準的な手順は、次のとおりです。

- ◆ ホスト接続を作成する。
- ◆ 接続を使用するログオンコンポーネントを作成する。
- ◆ ログオンコンポーネントを使用するログオン接続を作成する。
- ◆ ログオン接続を使用する 1 つまたは複数の 5250 端末コンポーネントを作成する。

接続の作成

この手順は簡単です。このガイドの第 2 章での説明に従って新しい接続リソースを作成します。ログオン接続で後に定義されたユーザ ID とパスワードを使用する場合でも、接続と同様にユーザ ID とパスワードを定義する必要があります。この操作は、次の手順でログオンコンポーネントを定義する場合に必要となります。他の方法としては、単に既存の接続リソースを使用することもできます。

ログオンコンポーネントの作成

➤ 5250 ログオンコンポーネントを作成する

- 1 Composer の **[File]** メニューから、**[New xObject]**、**[Component]**、**[5250 Logon]** の順に選択します。

New xObject Wizard のヘッダ情報パネルが表示されます。

Create a New 5250 Logon Component

A 5250 Logon Component manages a connection that can be used by multiple 5250 Terminal Components. Use this wizard to create a 5250 Logon Component. Enter a name and, optionally, a description for the component. The name will appear in the Composer Detail Pane and in choice lists when you are prompted for objects in Composer. The name may not contain the characters: \ / : ? " < > . | Names are case insensitive (i.e. MyObjectName is the same as myobjectname).

Name:
5250Sample

Description:

Purpose:
Input:
Output:
Remarks:

Help Back Next Cancel

- 2 接続オブジェクトの「名前」を入力します。
- 3 オプションとして、「説明」のテキストを入力します。
- 4 **[Next]** をクリックすると、「接続情報」パネルが表示されます。

Create a New 5250 Logon Component

Specify which Connection you wish to use for this Component or Service. To change any connection parameters, you must change them in the Connection Resource object or create a new Connection Resource of the same type with different parameters.

Connection: TN5250 to AS400 Port 23

Host or IP Address: AS400

Telnet Port: 23

Code Page: CP037

Screen wait (seconds): 60

User ID: GCARTER

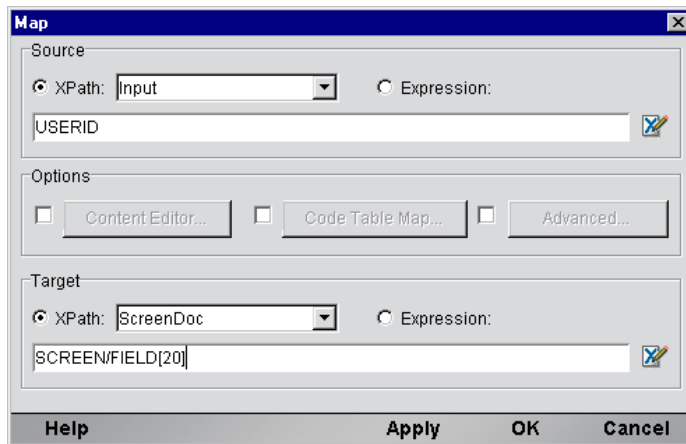
Password: *****

Terminal Type: BM-3179-2 (24 x 80)

DBCS Support: Default

Buttons: Test, Help, Back, Finish, Cancel

- 5 ドロップダウンリストで接続を選択します。
- 6 [Finish] をクリックすると、ログオンコンポーネントエディタが表示されます。
注記: アクションを記録するには、次の手順に従います。「LOGON」上にカーソルを合わせてから、記録をオンにします。完了すると、記録をオフにします。「KEEPALIVE」にカーソルを合わせてから、記録をオンにします。完了すると、記録をオフにします。「LOGOFF」にカーソルを合わせてから、記録をオンにします。完了すると、記録をオフにします。
- 7 このガイドの第4章で説明されている記録方法と同じ方法でホストにログインして、起動画面に移動するために、LOGON アクションを記録します。
- 8 このガイドの第4章の「5250 専用の Expression Builder 拡張」という節で説明されている特別な USERID 変数と PASSWORD 変数を代わりに使用するよう、ユーザ ID とパスワードを入力する LOGON Map アクションを編集します。

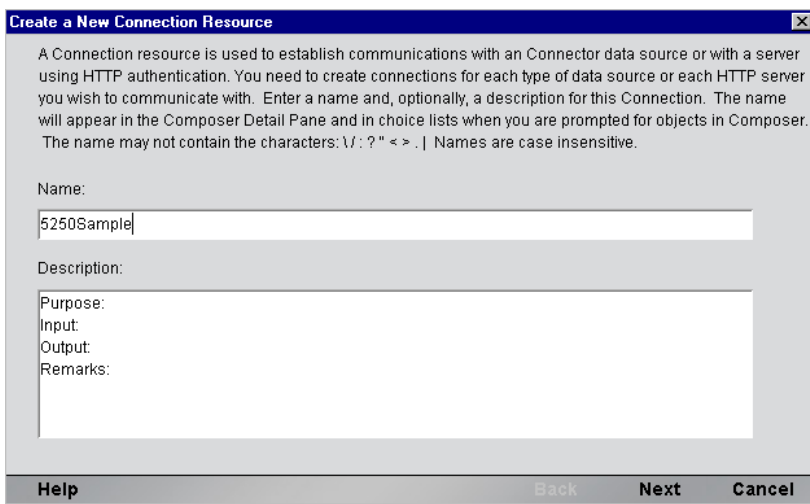


- 9 アクションモデルの KEEPALIVE セクションで、必要な SEND Key アクションを作成します (簡単な方法としては、既存の SEND key アクションをコピーして、貼り付けてから、送信されたキーコードを修正します)。
- 10 ホストを適切に終了するために、LOGOFF アクションを記録します。
- 11 ログオンコンポーネントを保存して、閉じます。

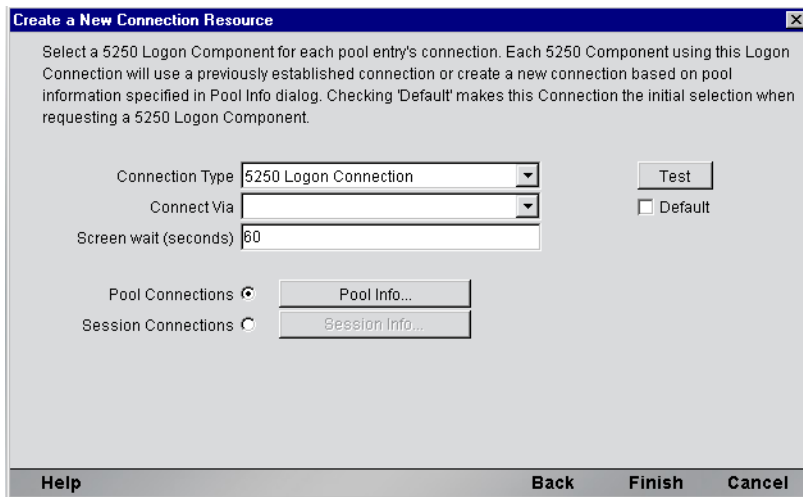
プール接続を使用したログオン接続の作成

➤ 5250 ログオン接続を作成する

- 1 Composer の [File] メニューから、[New xObject]、[Resource]、[Connection] の順に選択するか、またはアイコンをクリックします。New xObject Wizard のヘッダ情報パネルが表示されます。



- 2 接続オブジェクトの「名前」を入力します。
- 3 オプションとして、「説明」のテキストを入力します。
- 4 [Next] をクリックすると、「接続情報」パネルが表示されます。



- 5 [Connection Type]に対して、ドロップダウンリストから[5250 Logon Connection]を選択します。
- 6 [Logon Via] コントロールで、作成したログオンコンポーネントを選択します。

- 7 [Pool Info] ボタンをクリックすると、[Pool Info] ダイアログボックスが表示されます。

Pool size specifies the total number of connections that can be established. Keep Alive, Inactivity and Entry wait parameters set the timings associated with each connection. Selecting "Override UID/PWD" allows you to specify different logons. The userid and password from the base connection will be used if no override is specified. Specify Reuse Connection to verify that the proper ScreenDoc state is present before a connection can be reused.

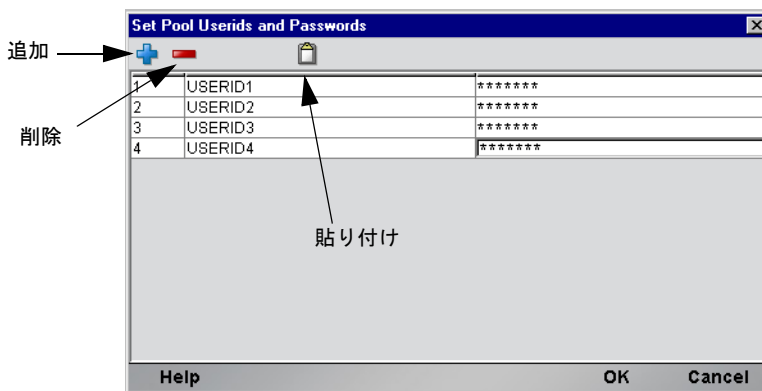
Pool size	2
Keep Alive (minutes)	2
Inactivity Lifetime (minutes)	60
Entry wait (seconds)	30
User ID	GCARTER
Password	*****
Override UID/PWD	<input checked="" type="checkbox"/> Set usersids...
Pool Device Names	<input type="checkbox"/> Set Device Names...
Use Sequential Connections	<input checked="" type="checkbox"/>
Retry discarded after (minutes)	1
Max retry discarded	3
Reuse connection only if expression is true	<input checked="" type="checkbox"/>
	oc.Xpath("SCREEN/FIELD(7)")="10051

Help OK Cancel

- 8 [Pool Size] に数値を入力します。これは、このプールで使用できるようにする接続の総数を表します。各接続に対して、後にユーザ ID とパスワードの組み合わせを指定する必要があります。
- 9 [KeepAlive]に期間を入力します。この数値(分単位)は、接続がアクティブであっても空き状態の場合に(つまり、5250 端末コンポーネントで使用されていない)、関連付けられているログオンコンポーネントで KEEPALIVE アクションを実行する頻度を表します。ここで入力する数値は、アクティブでない接続に対してホストで定義されたタイムアウト期間の値よりも小さくする必要があります。
- 10 [Inactivity Lifetime] に値を入力します。この数値は、接続を切断してから接続プールの非アクティブな部分に戻すまでに、アクティブな空き接続を使用可能な状態で維持する期間を分単位で表します。接続がプールで非アクティブな状態に戻されてから再びアクティブにされると、ログインしてホスト画面に移動する際にオーバーヘッドが発生することに注意してください。

- 11** [Entry Wait] に時間を秒単位で入力します。この時間は、すべてのプールエントリがアクティブで使用されている場合に、5250 端末コンポーネントが空き接続に対して待機する時間を表します。この時間に達すると、アプリケーションサーバに例外がスローされます。
- 12** [Override UID/PWD] をオンにすると、接続プールで使用するユーザ ID/パスワードの組み合わせを指定できます。このチェックボックスがオンの場合、[Set USERID] ボタンがアクティブになります。このボタンをクリックすると、[Set USERIDs and PASSWORDS] ダイアログボックスが表示されます。
- 13** 指定したプールのサイズになるまで **USERID/PASSWORD** の組み合わせの入力を繰り返し、[OK] をクリックします。(指定したサイズ以上に入力すると、自動的にその新しいサイズに調整されます。)

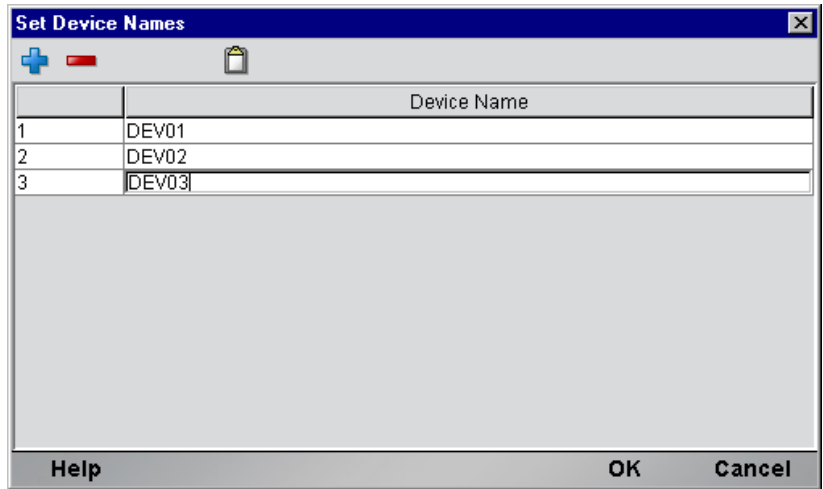
ツールバーには、空白の行を追加する [Add]、選択した行を削除する [Delete]、およびスプレッドシートの情報をコピーして表に貼り付ける [Paste] の 3 つのアイコンがあります。詳細については、次の注記を参照してください。



- 14** [Pool Device Names] のチェックマークをオンにすると、接続プールで使用するデバイス名を指定できます。このチェックボックスをオンにすると、[Set Device Names] ボタンがアクティブになります。このボタンをクリックすると、[Set Device Names] ダイアログボックスが表示されます。

注記: ログオンコンポーネントの接続の [Enable Telnet Environment] チェックボックスがオフになっている場合は、[Pool Device Names] チェックボックスと [Set Device Names...] ボタンの両方が無効になります。

- 15** 必要なデバイス名を入力し、[OK] をクリックします。



ツールバーには、空白の行を追加する[Add]、選択した行を削除する[Delete]、およびスプレッドシート情報をコピーして表に貼り付ける [Paste] の3つのアイコンがあります。詳細については、次の注記を参照してください。

注記： データをより迅速に入力する別の方法は、スプレッドシートからデータをコピーして、表に貼り付けることです。その場合、2列が選択されていることを確認してください。最初の列にはユーザIDが含まれており、2番目の列にはパスワードが含まれている必要があります。スプレッドシートを開き、この2列と必要な数の行をコピーします。表を開いたら、ツールバーにある [Paste] アイコンをただちに押します。同じ方法で、Microsoft Word/Excelドキュメントの表からデータをコピーすることもできます。

- 16** 連続的な接続を使用する場合は、[Use Sequential Connections] チェックボックスをオンにします。接続は、1、2、3のように、番号順で作成されます。
- 17** 破棄された接続の使用を試行する (プールに追加し直す) 分数を、[Retry discarded after] フィールドに数値を入力することによって指定します。
- 18** 破棄された接続の使用を試行する (プールに追加し直す) 最大回数を、[Max retry discarded] フィールドに数値を入力することによって指定します。

- 19** オプションとして、[**Reuse Connection Only if expression is true**] コントロールをオンにできます。このコントロールでは、新しい 5250 コンポーネントでアクティブな空き接続を再使用するたびに、起動画面が存在することを確認する式を入力できます。Composer サービスに関連していない状況では、起動画面がホストによって別の画面で置き換えられる可能性があります。たとえば、システム ABEND がホストに存在する場合、ログオンコンポーネントの起動画面はシステムメッセージ画面によって置き換えられることがあります。この式の作成方法の詳細については、このガイドの第 2 章の「システムメッセージの処理」という節を参照してください。この章の「5250 ログオン接続のパフォーマンスの最大化」も参照してください。

注記： 特定の画面が存在するかどうかを確認する場合に使用するカスタムスクリプトのサンプルを次に示します。画面が存在しない場合、スクリプトにより、不正な画面のためログオン接続が解放されることを示すメッセージがコンソールに記述されます。この関数は、[Pool Info] ダイアログボックスの [Reuse connect only if expression is true] コントロールから呼び出されます。

```
function checkValidLaunchScreen(ScreenDoc)
{
    var screenText = ScreenDoc.XPath("SCREEN").item(0).text
    if((screenText.indexOf("MENU") != -1 || screenText.indexOf("APLS") != -1) &&
        (screenText.indexOf("COMMAND UNRECOGNIZED") == -1 ||
         screenText.indexOf("UNSUPPORTED FUNCTION") == -1))
    {
        return true;
    }
    else
    {
        java.lang.System.out.println("Warning - Releasing logon connection at bad screen");
        java.lang.System.err.println("Warning - Releasing logon connection at bad screen");
        return false;
    }
}
```

20 [OK] をクリックして、接続情報パネルに戻ります。

21 [Finish] をクリックすると、ログオン接続が保存されます。

5250 ログオン接続のパフォーマンスの最大化

5250 コンポーネントが前の 5250 コンポーネントによって無効な画面に残された可能性のある接続で実行を開始しないようにするために、ログオン接続リソースでは、接続自体で起動画面が存在することを確認できます。これを実現するには、ログオン接続の [Pool Info] ダイアログボックスの [Reuse connection only if expression is true] オプションを使用します。ここで指定した画面テストは、5250 コンポーネントの実行が完了するたび行われます。テストに失敗した場合、exteNd Composer はただちにホストから切断し、その時点まで使用していたユーザ ID はホストに残される可能性があります。前に説明したように、ホストではこのユーザが最終的には停止されますが、ユーザが停止される前にもう一度ユーザ ID にアクセスした場合は、ユーザ ID がプールから破棄される可能性があります。したがって、プールサイズが小さくなり、その結果全体的なパフォーマンスも下がります。

[Reuse connection only if true] オプションを使用する別の理由は、このオプションでは画面に対して詳細なテストを実行して、正しい起動画面であるかどうかを確認できるためです。Map Screen アクションでは、画面の確認を実行しますが、端末データストリーム内のフィールド数のみが参照されます。大抵の場合は、これで十分ですが、異なる 2 つの画面にあるフィールドの数が同じ場合は、画面のコンテンツを検査する式ベースのテストを使用すると、さらに厳密な結果が得られます。そのため、この機能を常に使用することが推奨されます。

動的に作成したドキュメント / 要素と静的に作成したドキュメント / 要素の違い

一部の Composer アプリケーションでは、ユーザが各種のコントロール、監査、またはメタデータ、あるいはこれらすべてを XML ドキュメントに配置しなければならない場合があります。このドキュメントは、処理中の（つまり、情報ソースから作成された）実際の要素 / ドキュメントの追加となる場合もあれば、そうではない場合もあります。このドキュメントの構造およびデータが複数（100 以上）の Map アクションにより動的に作成された場合、コンポーネントのパフォーマンスおよびサービス全体に障害が発生する可能性があります。パフォーマンスを向上するには、事前に動的なコンテンツのないドキュメント構造の一部を作成します。その後、そのドキュメント構造を XML Interchange アクションを使用してランタイム時にサービスにロードし、動的な内容に対して Map アクションを保持します。この結果、場合によってはパフォーマンスが 30% ほども向上することがあります。

セッション接続を使用したログオン接続の作成

➤ 5250 ログオン接続を作成する

- 1 Composer の [File] メニューから、[New xObject]、[Resource]、[Connection] の順に選択するか、アイコンをクリックすることもできます。New xObject Wizard のヘッダ情報パネルが表示されます。

Create a New Connection Resource [X]

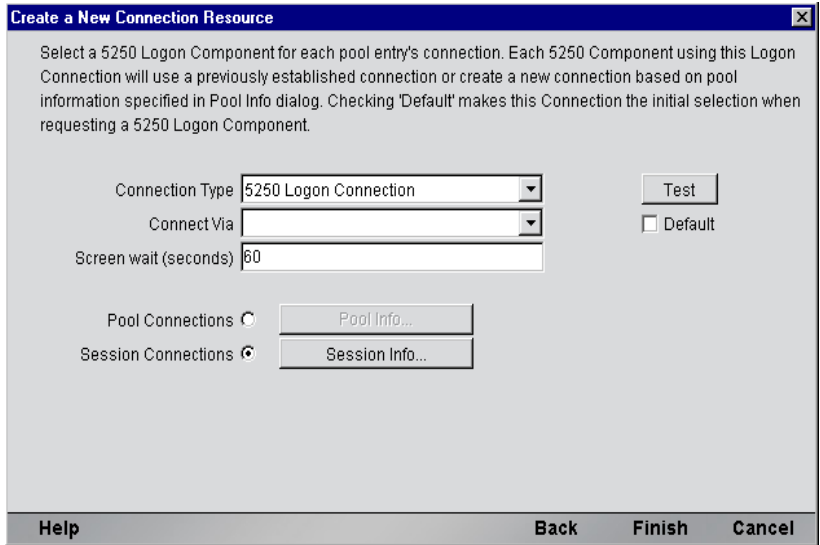
A Connection resource is used to establish communications with an Connector data source or with a server using HTTP authentication. You need to create connections for each type of data source or each HTTP server you wish to communicate with. Enter a name and, optionally, a description for this Connection. The name will appear in the Composer Detail Pane and in choice lists when you are prompted for objects in Composer. The name may not contain the characters: \ | : ? " < > . | Names are case insensitive.

Name:
5250Sample

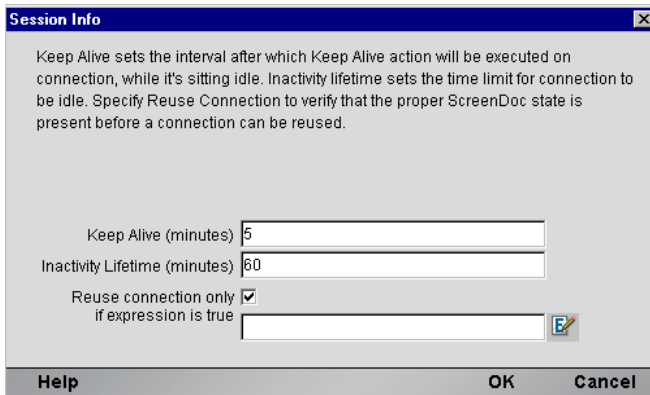
Description:
Purpose:
Input:
Output:
Remarks:

Help Back Next Cancel

- 2 接続オブジェクトの「名前」を入力します。
- 3 オプションとして、「説明」のテキストを入力します。
- 4 [Next] をクリックすると、「接続情報」パネルが表示されます。



- 5 [Connection Type]に対して、ドロップダウンリストから[5250 Logon Connection]を選択します。
- 6 [Connect Via] コントロールで、作成したログオンコンポーネントを選択します。
- 7 [Screen wait (seconds)] フィールドには、Map アクションペインに次の画面が到着するまで 5250 端末コンポーネントで待機する時間が秒単位で表示されます。
- 8 [Session Connections] ラジオボタン、[Session Info] ボタンの順にクリックします。



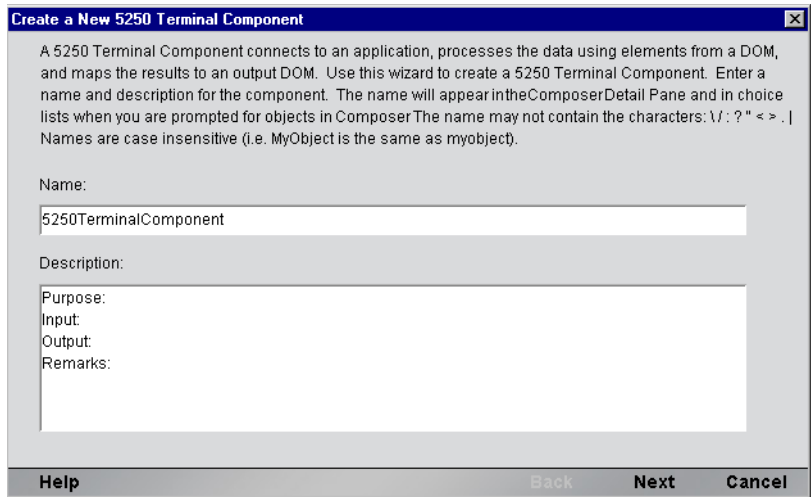
- 9 [Keep Alive (minutes)]の数値(分単位)は、接続がアクティブであっても空き状態の場合に(つまり、5250 端末コンポーネントで使用されていない)、関連付けられているログオンコンポーネントでKEEPALIVEアクションを実行する頻度を表します。ここで入力する数値は、アクティブでない接続に対してホストで定義されたタイムアウト期間の値よりも小さくする必要があります。
- 10 [Inactivity Lifetime (minutes)] の数値は、接続を切断してから接続プールの非アクティブな部分に戻すまでに、アクティブな空き接続を使用可能な状態で維持する期間を分単位で表します。接続がプールで非アクティブな状態に戻されてから再びアクティブにされると、ログインしてホスト画面に移動する際にオーバーヘッドが発生することに注意してください。
- 11 式が true のときにだけ接続を再使用する場合は、[Reuse connection only if expression is true] チェックボックスをオンにします。このチェックボックスをオンにすると、[Expression] フィールドが自動的に表示され、式アイコンをクリックして「if the expression is true」のダイアログボックスを開くことができます。

5250 端末コンポーネントの作成

これで、接続プールを使用できる 5250 ログオンコンポーネントを作成する準備ができました。ほとんどの場合、通常の 5250 端末コンポーネントの場合と同じようにコンポーネントを作成でき、唯一の違いは、New xObject Wizard で指定する接続となります。

➤ 5250 端末コンポーネントを作成する

- 1 Composer の [File] メニューから、[New xObject]、[Component]、[5250 Terminal] の順に選択します。New xObject Wizard のヘッダ情報パネルが表示されます。



- 2 コンポーネントの「名前」を入力します。
- 3 オプションとして、「説明」のテキストを入力します。
- 4 [Next] をクリックすると、XML プロパティ情報パネルが表示されます。
- 5 必要な入力テンプレートと出力テンプレートを選択して [Next] をクリックすると、接続情報パネルが表示されます。
- 6 作成したログオン接続を選択して [Next] をクリックすると、コンポーネントエディタが表示されます。
- 7 このガイドの第3章の手順に従って、コンポーネントを作成します。

5250 端末コンポーネントのパフォーマンスの最大化

ログオンコンポーネントの Logon アクションによって起動画面に達すると、この起動画面は接続を使用する 5250 端末コンポーネントに渡されます。その後、(実行終了時に) 5250 端末コンポーネントによって、画面ハンドラは起動画面に残されたままの状態になります。5250 コンポーネントが起動画面に達することなく終了した場合 (つまり、不正な画面でプールの接続が解放された場合)、その接続を使用する後続の 5250 コンポーネントすべてによって例外がスローされ、接続が使用できなくなる可能性があります。また、全体的なパフォーマンスが低下したり、コンポーネントの処理内でデータの整合性に問題が生じる可能性もあります。

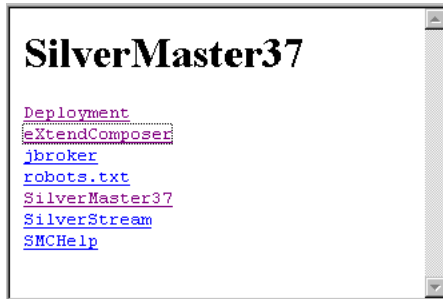
このためにも、起動画面が存在し、5250 コンポーネントで実行する最後のアクションが起動画面を確認する Map Screen になるようにします。コンポーネントの実行を独立して終了する可能性のある決定パスがコンポーネントに多数ある場合は注意が必要です。このような場合は、パスがそれぞれ Map Screen アクションで終了するようにします。

プールの管理

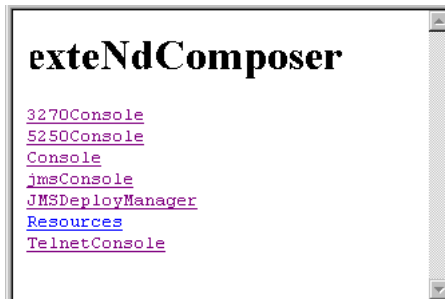
接続プールは、5250 コンソール画面を使用して管理できます。

➤ コンソールへのアクセス方法

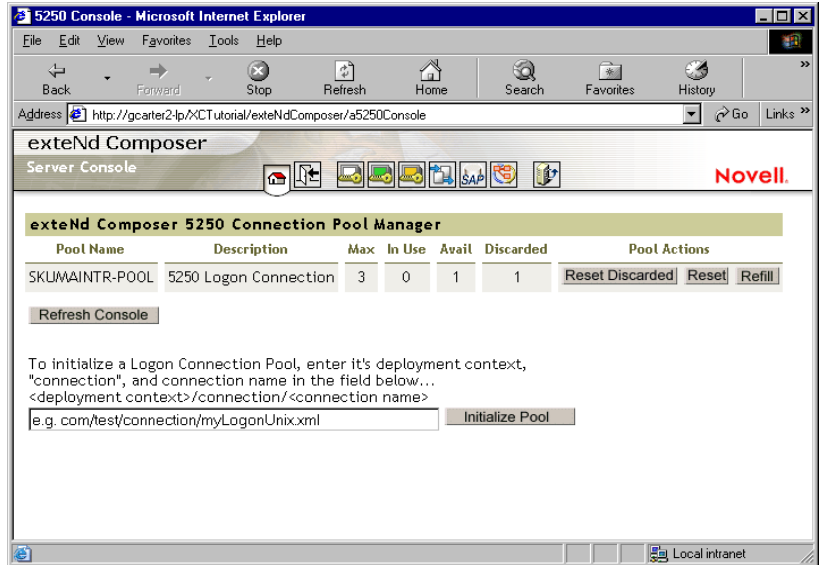
- 1 Webブラウザから<http://localhost/SilverMaster35>にアクセスし、サーバにログオンします。この例では、Novell サーバを使用します。



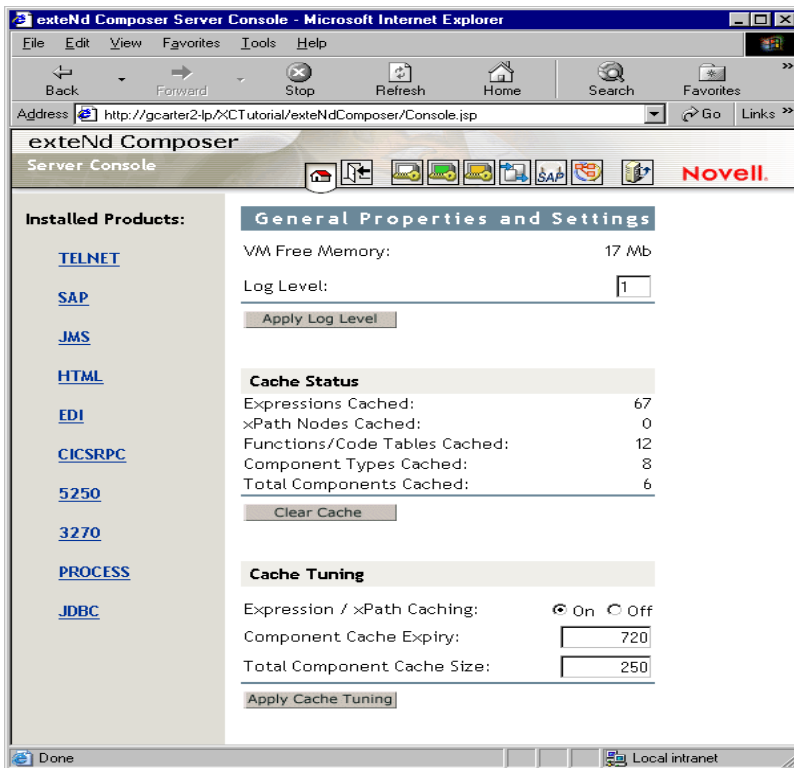
- 2 [exteNd Composer] をクリックすると、インストール済みの接続のリストが表示されます。



- 3 [5250] をクリックすると、コンソール画面が表示されます。フィールドにパスと名前を入力して、[SUBMIT] ボタンをクリックします。5250 Connect に対して接続がすでに存在している場合は、接続プール管理画面に表が表示されます。詳細については、該当するサーバガイドを参照してください。



exteNd Composer コンソールの左側の [Installed Products] リストで [5250] を選択して、exteNd Composer 5250 Connection Pool Managerを表示することもできます。



接続プール管理および配備済みのサービス

接続プールの管理画面には、5250 Connect を使用した接続の現在の状態が表示されます。画面には、プール名、接続の説明、プールでの最大接続数、最大使用接続数、使用可能な接続数、および破棄された接続数をリストした表が含まれています。また、接続をリセットできるようにするボタンがある 3 つの列も含まれています。

[Reset Discarded] ボタンは、破棄された接続をリセットし、表に反映します。[Reset Pool] ボタンは、使用可能な接続と破棄された接続をリセットし、表に反映します。[Refill Pool] ボタンは、プールを最大接続数まで再読み込みします。

表の下には [Refresh] ボタンがあり、このボタンをクリックすると、接続プールの現在の状態が表示されます。この下のフィールドでは、配備済みの lib ディレクトリへの相対パスを入力することによって、ログオン接続プールを初期化できます。この操作は、配備済みの jar を解凍しないと動作しません。終了すると [SUBMIT] ボタンをクリックします。

接続破棄の動作

接続プールにおけるパフォーマンス上の利点は、1 度に 1 人以上のユーザが単一のリソースやリソースのセットにアクセスできることです。ログオンコンポーネントでの接続の確立は、テーブルからユーザ ID とパスワードを選択することから開始します。接続に失敗すると、このユーザ ID とパスワードに対して接続が破棄され、接続が確立されるまで別の接続の確立が試行されます。接続に失敗しても、正常な接続の確立が妨げられるわけではありません。

5250 用 Connect では、不正なユーザ ID、タイムアウトしたパスワードなど理由を問わず、確立できなかった接続をすべて破棄し、他の接続を再使用することによって、確立できなかった接続以外に同じ問題が発生しないようにします。接続が使用不可能と判断された場合、5250 用 Connect では、「Logon connection in pool <Pool name> was discarded for User ID <User ID>」というメッセージをシステムログに書き込みます。

画面の同期

画面を同期すると、プールのユーザに特別な影響が生じます。たとえば、あるユーザが画面を元の状態に戻さずに接続から離れた場合、次のユーザは、予期しない状態の画面でセッションを開始することになるため、エラーが発生します。これを防ぐために、ユーザが接続プールで指定できる画面式が用意されています。5250 コンポーネントの最後のアクションは、正しいログオン画面がアクティブな状態でセッションを終了させるようにする正しい Send Attention Key アクションであることが重要です。

ユーザセッションの終了時に不正な画面があるかどうかをラインタイム時に確認する場合は、関数を実行するコンポーネントのアクションモデルの最後に Function アクションを含めます (次を参照)。

```
function checkValidReleaseScreen(ScreenDoc)
{
    var screenText = ScreenDoc.XPath("SCREEN").item(0).text
    if((screenText.indexOf("MENU") != -1 ||
screenText.indexOf("APLS") != -1) &&
        (screenText.indexOf("COMMAND UNRECOGNIZED") == -1 ||
screenText.indexOf("UNSUPPORTED FUNCTION") == -1))
    {
        return true;
    }
    else // Write error messages to
        // System.out and System.err:
    {
        java.lang.System.out.println("Warning - Releasing logon
connection at bad screen");
    }
}
```

```
java.lang.System.err.println("Warning - Releasing logon  
connection at bad screen");  
    return false;  
    }  
}
```

この関数では、画面テキストを確認し、最後の画面が正しくない場合に「false」を返します。画面に「MENU」や「APLS」が含まれており、「COMMAND UNRECOGNIZED」や「UNSUPPORTED FUNCTION」は含まれていない場合は、「true」が返されます。

A

Java コードページ

文字エンコーディングについて

exteNd Composer の文字エンコーディング変換機能は、使用されている Java VM と密接な関係があります。サポートされる文字エンコーディングは、Java 2 プラットフォームの実装によって異なります。Sun の Java 2 Software Development Kit, Standard Edition, v. 1.2.2 for Windows or Solaris および Java 2 Runtime Environment, Standard Edition, v. 1.2.2 for Solaris では文字エンコーディングがサポートされます。文字エンコーディングは、次の Sun の Web ページで参照できます。

<http://java.sun.com/products/jdk/1.2/docs/guide/internat/encoding.doc.html>

Sun の Java 2 Runtime Environment, Standard Edition, v. 1.2.2 for Windows には、米国版と国際版の 2 種類があります。国際版 (libi18n.jar ファイルを含む) では、両方の表にあるすべての文字エンコーディングがサポートされます。

B

5250 用語集

AID キー

5250 でサポートされているキーは次のとおりです。

表 B-1

5250 キー	PC キー
Enter	Enter
Clear	ESC
F1 から PF12	F1 から F12
F13 から F24	Shift+F1 から Shift+F12
PA1 から PA3	Ctrl+F1 から Ctrl+F3

Map Screen アクション

新しい端末データストリーム (TDS) 画面が受信されるアクションモデル内の場所を示す、編集できない特別なアクション。この画面を操作するアクションは、すべてアクションモデル内で Map アクションの Screen アクション行のサブオーディネートに配置する必要があります。

Multi Row 変換

DOM と 5250 画面の間で、5250 画面内のデータの行を表すマッピングに対する特別なアクション。また、このアクションでは、ScreenDom を変換して、画面の階層ビューを作成します。このアクションは、Multi Row Group アクション、Decision アクション、および他の Repeat アクションと同様に、Multi Row ウィザードによって作成されます。これらのアクションの編集時には特別な注意が必要です。

ScreenDoc

現在の 5250 画面表示を XML ドキュメントとして表す、5250 (および 3270) コンポーネントエディタウィンドウにある特別な DOM。

Send Attention Key

AID キーを押すとアクションモデルに表示されるアクション。

TDS

Terminal Data Stream (端末データストリーム)。

ネイティブ環境ペイン

実際の 5250 端末セッションのエミュレーションを表示する、5250 コンポーネントエディタ内のペイン。

フィールド

TDS に含まれるデータユニット。フィールドは、画面、データの項目、または表示されない特別な空白のフィールドに表示されるラベルである可能性があります。各フィールドには、表示方法と領域を変更できるかどうかを決める固有の属性があります。

C

テスト

アニメーションテストと配備テストの環境的相違

Composer でのアニメーションテストと配備テストでは、かなり環境的な違いがあります。これらのテストはいずれも作成したコンポーネントやサービスを適切に検証するために必要です。違いについては、次の表に詳しく説明します。

表 C-1

	Composer でのテスト	配備テスト
OS	Win98、WinNT、または Win 2000	WinNT、Win2000、または Sun Solaris
プラットフォーム	JRE (Java Runtime Environment)	フェイルオーバー、セキュリティ、接続 MGT などに対する JRE サポートを完全に備えたアプリケーションサーバ
コンポーネントまたはサーバの起動	Composer から直接的	サービストリガによるのみ (つまり、配備サーブレットまたは EJB)
xObject アクセス	ディスクファイルから	アプリケーションサーバ内の JAR ファイルから
ランタイムコンテキスト	個々のコンポーネント、またはサービス内で実行されているコンポーネントをテスト	常にサービス内から
サービス入力およびコンポーネント入力	入力ドキュメントは、ローカルのサンプル XML ドキュメントだけでなく、他のサービスやコンポーネントからの DOM に由来することが多い	入力ドキュメントは、サービストリガ、または他のサービスやコンポーネントからの DOM を介して、サービスとコンポーネント内に渡される

	Composer でのテスト	配備テスト
次に対するプロジェクト変数: * ログファイルのパス * DTD URI * *XSL URI * メール送信サーバ * XML 交換 URI の変更	通常はローカルマシン上の場所を指します (ただし、サーバや Web 上の場所でも可)。	プロダクションサーバおよび Web 上の場所を指します。
テストツール	Log アクション以外に、ダイアログボックス (ECMAScript alert() 関数) を使用してランタイム値を表示することも可能	ダイアログボックスは使用できません。
JDBC 接続	サーバ接続プールを使用しない - テストデータベースを使用している可能性がある	サーバ指定の接続プールを使用する - 製品データベースを使用している
HTTP 接続	ローカルマシンまたはテストサーバを指している可能性がある	テストサーバまたはプロダクションサーバを指している
3270 接続	通常、システムのテスト、およびユーザ ID とパスワードのテストを指している	製品システム、および製品ユーザ ID とパスワードを指している
5250 接続	通常、システムのテスト、およびユーザ ID とパスワードのテストを指している	製品システム、および製品ユーザ ID とパスワードを指している
CICS(ECI) - RPC 接続	通常、システムのテスト、およびユーザ ID とパスワードのテストを指している	製品システム、および製品ユーザ ID とパスワードを指している

D

予約語

次の用語は、5250 対応の **exteNd Composer Connect** の予約語であるため、ユーザ作成ラベルまたはユーザ作成スクリプトでは使用しないでください。

- ◆ USERID
- ◆ PASSWORD
- ◆ PART
- ◆ MENU
- ◆ AID key
- ◆ SEND ATTENTION KEY
- ◆ MULTIROW

数字

- 5250 セッションの記録 47
- 5250 セッション、記録 47
- 5250 119
- 5250 (TDS) Connect 13
- 5250 固有
 - [Component] メニュー 39
 - [View] メニュー 39
 - コンテキストメニュー 40
 - メニューバー項目 39
- 5250 コンポーネント
 - 概要 14
 - 作成で使用する一般的な手順 17
 - 作成の前に 31
 - 実行 80
 - 新規作成 31
- 5250 コンポーネントエディタ
 - アプリケーションの作成 16
 - ウィンドウについて 34
 - はじめに 17
- 5250 専用
 - Expression Builder 拡張 46
 - Map Screen 44
 - Multi Row 45
 - Send Attention Key 45
 - アクション 44
- 5250 端末セッションパフォーマンス 99
- 5250 ネイティブ環境ペイン 35
- 5250 の XML テンプレート 29
- 5250 ログオンコンポーネント 103
- 5250 ログオン接続のパフォーマンスの最大化 119
- 5250 端末コンポーネントのパフォーマンスの最大化 123

A

- AID キー、定義 131
- Alias アクションの追加について 57

C

- COBOL 65

D

- Data Exchange アクション 93
- DOM 34

E

- EBCDIC 13
- ECMAScript 92
- exteNd Connect 12

G

- GORD 17

I

- [Iterate by attention key] 76

K

- KeepAlive アクションを使用したパフォーマンスの最大化 108

L

- LOGOFF アクションのパフォーマンスの最大化 109

M

- Map Screen アクション、定義 131
- MENU 17
- Multi Row
 - 5250 専用 45
 - ウィザードについて 59
 - ウィザードを使用したデータの出力 71
 - ウィザードを使用したデータの入力 60, 64
 - サンプルについて 60
 - データの出力に使用するアクションの追加 77, 78
 - データの入力に使用するアクションの追加 68

Multi Row 変換、定義 131

P

PART 17

Pool Device Names 116

R

Repeat アクション 94

S

ScreenDoc 34, 36

ScreenDoc DOM

機能 36

動作 36

ScreenDoc、定義 132

Send Attention Key

概要 45

定義 132

T

TDS、定義 132

Telnet ポート 20

U

Unicode 22

X

XSL 14, 134

あ

アクション

概要 43

基本および高度の使用 92

アクションの削除 59

[Action] メニュー 92

アクションモデル

以前の記録の編集 51

新しいアクションの追加 55

アテンションキー 13

アテンションキー、編集 54

アニメーション

実行 81

ツールの使用 81

アニメーションテストと配備テストの環境的

相違 133

アプリケーションの作成 16

う

ウィザード

Multi Row 59

データの出力に使用 71

データの入力に使用 60, 64

え

エラーおよびメッセージ 94

か

カスタムスクリプト、作成 30

画面の同期 127

画面フィールド数の変更 95

き

キーボードサポート 36

既存のアクションの変更 52

基本的なアクション 92

こ

高度なアクション 92, 93

コードテーブルマップ、作成 30

コードページ

サポート 22

文字エンコーディング 129

コンポーネントエディタウィンドウ 34

さ

サンプルトランザクション 17

す

スタイルシート

適用 91

ネイティブ環境ペインでの使用 91

スタイルシートリソース

作成 26

せ

セッション接続を使用したログオン接続の作成 120

接続情報パネル 33

接続破棄の動作 127

接続プールアーキテクチャ 100

接続プール管理および配備済みのサービス 126

接続プールの作成 110

接続リソース

5250 での作成 18

た

端末データストリーム 13

と

動的に作成したドキュメント / 要素と静的に作成した
ドキュメント / 要素の違い 119

トランザクション 95

トランザクション、サンプル 17

ね

ネイティブ環境ペイン 35

スタイルシートの使用 91

ネイティブ環境ペイン、定義 132

ふ

ファンクションキーマッピング 36

フィールド、定義 132

プールの管理 124

へ

別名 67, 75

ゆ

ユーザ ID 46

ろ

ログオンコンポーネントを使用したパフォーマンスの
最大化 106

ログオン接続の作成 113

