Documentation Project Plan

Open Enterprise Server 2 SP3

January 18, 2012

Novell.

Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the Novell International Trade Services Web page (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2010 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

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

Online Documentation: To access the latest online documentation for this and other Novell products, see the Novell Documentation Web page (http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Plan **7**

1 About the OES 2 SP3 Project **9**

1.1 Value Proposition . . . 9
1.2 Market . . . 9
1.3 Audience . . . 9
1.4 Delivery Methods . . . 10
1.5 Project Plans and Documents . . . 10
1.6 Localization . . . 11

2 Strategy for Meeting Project Objectives **13**

2.1 Documentation Project Objectives . . . 13
2.1.1 General . . . 13
2.1.2 Common Proxy User . . . 14
2.1.3 eDirectory Reconfiguration . . . 14
2.1.4 Other New Features . . . 15
2.1.5 Deployment Best Practices . . . 15
2.2 Information Accessibility . . . 15
2.3 Delivering the Doc . . . 15
2.3.1 During SP3 Development . . . 16
2.3.2 When SP3 Is Released . . . 16

3 Documentation Deliverables and Associated Source Files **17**

3.1 Deliverable Types . . . 17
3.2 SVN Storage Locations . . . 17

4 Team and Assignments **19**

5 Communication **21**

5.1 Project Coordination Calls . . . 21
5.2 Documentation Project Manager to the Documentation Team . . . 21
5.3 Documentation Project Manager to the Extended Core Team and Its Members . . . 21
5.4 Writers to Documentation Project . . . 22
5.5 Writers to Their Teams and Reviewers . . . 22
5.6 Reviewers to the Documentation Team . . . 22

6 OES 2 SP3 Documentation Schedule **23**

7 Status Reporting Requirements **25**

7.1 Hot Issues . . . 25
7.2 Documentation Development Status . . . 25

8 Production Plan **27**

8.1 Engineering Handoffs . . . 27
8.2 Publishing on Novell.com . . . 27
8.2.1 PDFs Are Always Included . . . 27
8.2.2 Publication Schedule . . . 27
8.2.3 Handing Off a Guide for Publication . . . 28
8.3 On-Demand Publishing . . . 28
8.3.1 The Needs . . . 28
8.3.2 The Solution . . . 28
8.3.3 Submitting a Guide for Publication . . . 28

9 QA Plan **29**

9.1 Technical Reviews . . . 29
9.2 Peer Reviews . . . 29
9.3 Editing Reviews . . . 30
9.4 Graphics Reviews . . . 30

10 Doc Testing and Feedback Planning **31**

10.1 About Documentation Testing . . . 31
10.1.1 Understanding Doc Testing as Defined for OES 2 SP3 . . . 31
10.1.2 When Should Testing Be Scheduled? . . . 31
10.1.3 Who Can Schedule Doc Tests? . . . 32
10.1.4 How Is Feedback Provided? . . . 32
10.2 Scope of OES 3 SP3 Documentation Testing . . . 32

11 Instructions for Documentation Testers (FAQs) **33**

11.1 What Is Documentation Testing in OES 2 SP3? . . . 33
11.1.1 What OES Doc Testing Is . . . 33
11.1.2 What OES Doc Testing Is Not . . . 34
11.2 When Do I Submit Feedback? . . . 34
11.3 What Pieces Should Be Tested? . . . 34
11.3.1 Focus on the HTML Guides . . . 34
11.3.2 Don’t Emphasize Context-Sensitive Help . . . 34
11.3.3 Ignore Sections Under Construction . . . 35
11.4 What Are the Feedback Mechanisms for Each Documentation Type? . . . 35
11.5 Why Not Use Bugzilla for HTML Feedback? . . . 36
11.6 What Should I Report? . . . 36
11.7 Are There Things I Should Not Report? . . . 37
11.8 How Do I Report Missing ECO and New Feature Items? . . . 37
11.9 What About Issues That Originate in Software Bugs? . . . 37
11.10 How Do I Check the Status of My User Comments? . . . 38
11.11 What Do the User Comment Severity Levels Mean? . . . 38

12 User Comments **39**

12.1 Preparing for User Comment Submission . . . 39
12.2 OES User Comment Goals . . . 39
12.3 Evaluating Suggestions for Rewrites . . . 40
12.4 Handling User Comments . . . 40

13 Localization Plan **41**

13.1 Overview of Help Localization Processes . . . 41
13.2 Keeping Localization Costs As Low As Possible . . . 42

A Working with Subversion **45**

A.1 Gaining Access to the Repository . . . 45
A.2 Installing the TortoiseSVN Client . . . 46
A.3 Creating a Work Directory on Your Local Drive or Home Directory . . . 46
A.4 Understanding Important TortoiseSVN Menu Options . . . 47
A.4.1 Add . . . 47
A.4.2 Checkout . . . 47
A.4.3 Delete . . . 49
A.4.4 SVN Commit . . . 49
A.4.5 SVN Update . . . 49
A.5 Populating Your Work Directory . . . 49
A.5.1 Checking Out Only the OES 2 Guides You Need . . . 49
A.5.2 Checking Out the Help Files You Need . . . 50
A.5.3 Checking Out the Entire OES 2 SP3 Project . . . 50
A.5.4 Checking Out Other Products you link to . . . 51
A.6 product.ini FAQ . . . 52
A.6.1 What is the function of the product.ini file? . . . 52
A.6.2 Where is the product.ini file located? . . . 52
A.6.3 Is a product.ini file required in each book parent directory? . . . 52
A.6.4 I only link to sections within my own book. What do I need? . . . 52
A.6.5 What if I link to other books in the OES 2 doc set? . . . 53
A.6.6 Why must the idfiles directory be included in each Frame book? . . . 53
A.6.7 I have links to books in other products. What do I need to do to link to these other products?53
A.7 About the Standardized product.ini File . . . 54
A.8 Always Use TortoiseSVN Options in Windows Explorer . . . 54
A.9 Accessing the TortoiseSVN Documentation . . . 54

B Maintaining the OES Online TOC Pages **55**

B.1 Overview . . . 55
B.1.1 Historical Background . . . 55
B.1.2 The Current OES 2 TOC . . . 57
B.1.3 About the TOC Template . . . 61
B.1.4 About the TOC Macros . . . 62
B.2 Working in the TOC Book . . . 62
B.2.1 Authoring a Standard TOC Topic (Chapter) . . . 62
B.2.2 Authoring Chapters with No Subtopics . . . 66
B.2.3 Authoring an A–Z Chapter . . . 67
B.2.4 Focus on Helping OES Administrators . . . 67
B.2.5 Linking Guidelines . . . 68
B.2.6 Alphabetizing the TOC Tables . . . 70
B.2.7 Preserving Link Target Integrity Is Critical . . . 70
B.3 Producing the TOC . . . 70
B.4 The OES TOC Is Also an Important Tool . . . 72
B.4.1 Aiding the Migration from NetWare to OES . . . 72
B.4.2 Monitoring for Consistent and Complete Coverage . . . 72
B.4.3 Providing Just-in-Time Information . . . 73
B.5 Looking Forward to Ponderosa . . . 73

C Maintaining the OES Readme **75**

D Profiling in OES 2 SP3 **77**

D.1 Historical Perspective 77
D.2 Why the Proposed Change? 77
D.3 Open Enterprise Server Is a Unique Novell Product 77
D.3.1 Size 78
D.3.2 Service Diversity and Complexity 78
D.3.3 Link Density 78
D.3.4 Flagship Status 78
D.4 Challenges to Overcome 78
D.5 Plan for Overcoming 79

E Documentation Updates **81**

About This Plan

This is the documentation project plan for OES 2 SP3.

Audience

The plan is the primary project reference for writers assigned to the OES 2 SP3 project.

It is also available to others who have a vested interest in it, such as documentation managers, project managers, product managers, marketing managers, and documentation production personnel. And one chapter is specifically written to documentation testers (see Chapter 11, "Instructions for Documentation Testers (FAQs)," on page 33).

Feedback

If you have concerns about this plan, please use the User Comments feature to log them.

1 About the OES 2 SP3 Project

The following product details have implications for the documentation.

- Section 1.1, "Value Proposition," on page 9
- Section 1.2, "Market," on page 9
- Section 1.3, "Audience," on page 9
- Section 1.4, "Delivery Methods," on page 10
- Section 1.5, "Project Plans and Documents," on page 10
- Section 1.6, "Localization," on page 11

1.1 Value Proposition

The follow are project goals for OES 2 SP3 as identified by Marketing and Project Management:

It is critical that our NetWare customer base see SP3 as a true NetWare replacement. Therefore, there are two primary points of emphasis.

- **Stability:** SP3 must be a very stable release. The test cases are to be thorough and complete, and all testing is to be completed with no shortcuts or compromises.
- **Deployability:** All true migration inhibitors are to be addressed so that all customers can begin in earnest with replacing their NetWare servers with OES 2 SP3.

1.2 Market

The OES 2 market is unchanged from previous releases. It targets IT managers and business technology buyers who are considering replacing their Windows or NetWare infrastructure for workgroup application services.

These customers want a commercially supported Linux distribution that provides

- Ease of deployment
- Ease of management
- Identity integration with services.

1.3 Audience

OES 2 SP3 has the following target audiences, listed in order of priority.

1. NetWare administrators who need to transition their NetWare servers and Novell services to OES 2 Linux.

2. Network administrators (UNIX, Windows, Linux) who need to implement OES 2 services on their networks.
3. OES 2 administrators who need the new features and bug fixes in OES 2 SP3.

1.4 Delivery Methods

Customers can obtain OES 2 SP3 through the methods outlined in Table 1-1.

Table 1-1 *Methods of Delivery*

Distribution Method	Description
Physical Media	A set of physical CDs containing: ◆ SLES 10 SP3 ◆ OES 2 SP3 ◆ Identity Manager 3.6.1 Bundle Edition
Downloadable ISO Images and Executable Files	A set of images that allow the creation of CDs and DVDs that contain: ◆ SLES 10 SP2 ◆ OES 2 SP3 ◆ Identity Manager 3.6.1 Bundle Edition Executable and other files that install: ◆ Novell Client software for Windows distributions, including XP and Windows 7. ◆ Novell's Linux workstation distributions.

1.5 Project Plans and Documents

Core team plans and documents include the following:

- **Project Management:** The OES 2 SP3 core team manages the product using the OES 2 SP3 Teaming Site (https://teaming.innerweb.novell.com/ssf/a/c/p_name/ss_forum/p_action/1/action/view_ws_listing/binderId/73544).
- **Product Requirements:** Versions of the PRD are posted on the teaming site's PRD page (https://teaming.innerweb.novell.com/ssf/a/do?p_name=ss_forum&p_action=1&newTab=ssNewTabPlaceHolder&action=view_folder_listing&binderId=73838)

  All PRD requirements have corresponding entries in FATE (https://fate.novell.com).

  To see the SP3 requirements, do the following:

  1. Access the web.fate site (https://fate.novell.com).
  2. If prompted to log in, enter your novell.com username and password (the same as you use for Bugzilla access).
  3. Click *Browse* in the page banner.
  4. In the *Products* list, scroll down and select *Novell Open Enterprise Server 2.0.3 (OES 2 SP3)*.

5. Click *Search*.
6. Click the *ID* or *Title* entry to see requirement details.

- **Notes and Minutes:** Core team minutes, lists of personnel and their assignments, and other TWiki-based documents are posted on the OES 2 SP3 Twiki (https://twiki.innerweb.novell.com/bin/view/OES/OeS2Sp3).

1.6 Localization

Localization costs for OES 2 SP3 must be kept at a minimum.

For information about the documentation localization plan, see Chapter 13, "Localization Plan," on page 41.

2 Strategy for Meeting Project Objectives

OES 2 SP3 documentation adds to what was delivered for previous OES 2 releases as needed to meet the project's Value Proposition.

The guiding principle is to always remember that customers access documentation to complete a task or understand a concept associated with a task they need to perform.

The documentation strategy is defined in the following sections:

- Section 2.1, "Documentation Project Objectives," on page 13
- Section 2.2, "Information Accessibility," on page 15
- Section 2.3, "Delivering the Doc," on page 15

2.1 Documentation Project Objectives

The goals and objectives for OES 2 SP3 documentation are outlined below.

The following priority definitions apply:

- High = Mandatory
- Medium = Make every reasonable effort to complete as time permits
- Low = Desirable, but only do when it's not possible to work on High or Medium priority objectives.

2.1.1 General

Ensure that documentation is consistent, complete and up-to-date.

Objective	Owner	Priority
Determine the scope of the work by using the DPP	Writers, Shruthy, Larry	H
Comply with the OES 2 Project Style guidelines and content model as specified in the *OES2 SP3: Style Guide* (unchanged at this point for SP3).	Writers, Editors	H
Use at least one other source besides engineers to review/test the documentation (for example, testers, tech support, Beta customers, education).	Writers	H
Incorporate all posted User Comments for guides that are actively maintained.	Writers	H
"Scrub" the Readme. (identify and remove resolved issues. Make sure that caveats and issues that won't be fixed are given to the appropriate writers for inclusion in the product guides.)	Shruthy/Larry	M
Review all new UI, especially the installation, configuration, and migration interfaces, and help ensure that information is accurate and usable.	Writers	M
Become acquainted with other information produced for your feature/product, such as marketing flyers, white papers, TIDS, Cool Solutions articles, and education courses. Consider using the most informative and helpful of them as resources for content or as link targets.	Writers	L

2.1.2 Common Proxy User

Ensure that the information for understanding and deploying the common proxy user is accurate and complete.

Objective	Owner	Priority
Modify the proxy user information in "System User and Group Management in OES 2 SP3" in the *OES 2 SP3: Planning and Implementation Guide* to include information about understanding, planning and deploying the common proxy user. Point to additional information in the *OES 2 SP3: Installation Guide* and the various service guides as appropriate	Larry	H
Include installation and deployment instructions for the common proxy user in the *OES 2 SP3: Installation Guide*.	Larry	H
Include troubleshooting information about the common proxy user in the *OES 2 SP3: Planning and Implementation Guide*.	Larry'	H
Include installation and deployment caveat information for the common proxy user in the *OES 2 SP3: Planning and Implementation Guide*.	Larry	H
Include service-specific planning, deployment, and troubleshooting information for the common proxy user in the various service guides, and send the link information to Larry for inclusion in the *OES 2 SP3: Planning and Implementation Guide*.	Writers	H

2.1.3 eDirectory Reconfiguration

<Coming soon.>

2.1.4 Other New Features

Ensure that the information for understanding and deploying all new features is accurate and complete.

Objective	Owner	Priority
Document all new service features in the various service guides.	Writers	H
Supply Larry with a brief description of each new feature and then make sure that the "What's New or Changed" section in the *OES 2 SP3: Planning and Implementation Guide* includes a brief description of each new feature and a link to the appropriate service guide.	Writers	H
Receive new feature descriptions from service writers and teams and include in the "What's New or Changed" section in the *OES 2 SP3: Planning and Implementation Guide*.	Larry	H

2.1.5 Deployment Best Practices

Identify areas of major focus for deployment best practices and ensure that at least three of them are included in new P&I guide sections.

Objective	Owner	Priority
Work with Nicel and the PMs to identify which features are most problematic for customers to deploy and who can help identify appropriate deployment best practices for each.	Larry	H
Create Best Practices sections in the *OES 2 SP3: Planning and Implementation Guide* for at least the top three areas identified above.	Larry	H

2.2 Information Accessibility

The documentation for OES 2 SP3 is accessed through the same information categories (topics) that were used for OES 2 SP2. Subcategories (subtopics) help users navigate to the links for the information they are seeking.

Each category has a page that is accessed through links on the OES 2 Documentation Web site (http://www.novell.com/documentation/oes2). Categories are listed alphabetically, starting with "access, authenticate, log in" in the second and third columns in the top portion of each page. For more information, see Appendix B, "Maintaining the OES Online TOC Pages," on page 55.

An enhancement request has been logged in Bugzilla (Bug #629225) for enabling the User Comment features on each page.

2.3 Delivering the Doc

The team delivers documentation in the following ways.

2.3.1 During SP3 Development

While SP3 is developed, there are two sets of customers, each with different needs to meet.

- **Beta Customers:** Can access documentation specific to each Beta release on the Documentation Beta Web site (http://www.novell.com/documentation/beta/oes2).

  IMPORTANT: To help customers focus on the services featured in each release, we will publish only those admin guides, etc. that are pertinent to the services featured in the current and past Beta releases.

  The complete documentation set for OES 2 SP3 will not be published until Beta 3 (public Beta).

- **Writers and Internal Reviewers:** Writers can refer reviewers to the novell.com staging server (http://wwwtest.provo.novell.com/documentation/oes2/index.html) for
  - Verifying that Bugzilla defects have been addressed
  - Participating in technical reviews
  - General viewing of how the SP3 documentation set is evolving

  In contrast to the limited documentation posted on the Beta site, the staging server will host the most recent versions of all the OES 2 SP3 documentation.

  IMPORTANT: Using the wwwtest staging server and the User Comments feature for internal reviews is the standard review practice for the OES 2 SP3 documentation team.

2.3.2 When SP3 Is Released

Customers access documentation deliverables in two places:

- **Novell.com:** When SP3 is released, all guides will be available in HTML and PDF formats on the OES 2 Documentation Web Site (http://www.novell.com/documentation/oes2). ESD documents are posted online with the OES 2 SP3 ISO image files and other components on download.novell.com (http://download.novell.com).
- **Product Builds:** Local help files, man pages, and pointer readme files are included in product builds.

3 Documentation Deliverables and Associated Source Files

All documentation deliverables for OES 2 SP3 are listed in the OES 2 SP3 DPP (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_proj_index.jsp?pid=36484).

- Section 3.1, "Deliverable Types," on page 17
- Section 3.2, "SVN Storage Locations," on page 17

3.1 Deliverable Types

For project management purposes, documentation deliverables are classified in the following general deliverable types.

- **Manuals and Guides:** These include administration guides, user guides, quick start guides, and so on. Together they form the bulk of the online documentation set published on the Web.
- **Help Files:** These are tied to a software help system of some kind, and include the Novell traditional C-Worthy help, various types of HTML-based help, Java help, and so on.
- **Localization Handoff Files:** These are copies of other documentation deliverables that have been identified as requiring localization.
- **Readmes and Release Notes:** This includes readmes and release notes that are distributed on the media, included with the product, and/or published with the online documentation.
- **ESD (Electronic Software Delivery) Files:** Novell publishes a standard set of small documentation files with each downloadable software package. ESD files are required for the OES 2 ISO images as well as smaller, bundled components that are distributed separately.

3.2 SVN Storage Locations

Source and output files are stored by deliverable type in the SVN repository as outlined in the following table:

Deliverable Type	Location in the Open_Enterprise_Server SVN Directory Structure	Notes
Manuals and Guides	`/oes2_sp3/`frame/*guide_name*	These directories include the Frame source, authoring support, and various output files. If file sets produce both guides and help files, they are stored here. for more information, see the Authoring Guide (http://www.novell.com/documentation/beta/docguides/authprodguide/data/bktitle.html).
Help Files	`/oes2_sp3/help/`*`help_type`*/*help_name*	All help source and output files are stored here. This includes Frame source files for help-only deliverables.
Localization Handoff Files	`/oes2_SP3/handoff_loc`	The sibling directory structure to `handoff_loc`, i.e. `frame`, etc. is repeated in this structure as required to organize various localization handoff files.
Readmes	`/oes2_SP3/readme/`	OES 2 static and pointer readmes are stored here. The product readme is in the frame directory because it is a complete guide that links to OES 2 and other guides.
ESD Files	`/oes2_SP3/esd`/*other*/*directories*/*as_needed*	ESD files are generally specifically-formatted HTML files as dictated by Novell Operations. Directories and sub-directories are named to reflect the software components that are covered by the ESD files.

4 Team and Assignments

All team members are listed on the OES 2 SP3 DPP (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_proj_index.jsp?pid=36484).

Writers are also listed with their team assignments in the OES 2 SP3 Address Book (https://twiki.innerweb.novell.com/bin/view/OES/OeS2SP3AddressBook).

However, managing the documentation project requires a slightly different breakdown that includes key guides and bundled components that are not listed in the other places.

To help you quickly identify which writer has what, documentation team members and their assignments are listed in Table 4-1.

Table 4-1 *Components and Assigned Writers*

Component	Writer
AFP	Shruthy Devendra
Arch and Version Services	Shilpa Bandekar
CASA	indumathi Indumathi
Certificate Server	
CIFS	Shruthy Devendra
Clustering	Catherine Craft
ConsoleOne	(unassigned)
Customer Center	(unassigned)
DHS/DHCP	Shruthy Devendra
Distributed File Services	Shilpa Bandekar
Domain Services for Windows	Shaju John
Dynamic Storage Technology	Catherine Craft
eDirectory	Menaka Lekshmiganthan
File Systems Management	Catherine Craft
FTP	Shruthy Devendra
IDM 3.6.1 Bundle Edition	Jagadeesh Katiperi
iFolder 3.8.1	Shaju John
iManager	Jagadeesh Katiperi
Installation	Larry Tomkinson

Component	Writer
iPrint	Archana Singh
Kerberos KDC	Jagadeesh Katiperi
Getting Started Guide	Larry Tomkinson
Linux Tips	Larry Tomkinson
Login Scripts	
LUM	
Migration Tools	Shilpa Bandekar
NCP	Catherine Craft
NDK	Jagadeesh Katiperi
NetStorage	
NICI	
NMAS	
Novell Client	Triveni Mahesh
NRM	Catherine Craft
NSS	Archana Singh
Online Readme	Shruthy Devendra
OpenWBEM	Catherine Craft
Password Management	
Plan and Implement	Larry Tomkinson
QuickFinder	Shaju John
Remote Management (rconsole)	Shaju John
Samba	Larry Tomkinson
SBCON	Shilpa Bandekar
SCMT 1.2	Shilpa Bandekar
SMS	Shilpa Bandekar
Storage Disks and Devices **	Catherine Craft
Storage Overview	Catherine Craft
Traditional File Services **	Catherine Craft
Virtualization	SUSE (Larry Tomkinson for installation guide pieces)
Web Overview	

** Bug Fixes Only

5 Communication

The communication plan for OES 2 SP3 includes the following communication paths and requirements:

- Section 5.1, "Project Coordination Calls," on page 21
- Section 5.2, "Documentation Project Manager to the Documentation Team," on page 21
- Section 5.3, "Documentation Project Manager to the Extended Core Team and Its Members," on page 21
- Section 5.4, "Writers to Documentation Project," on page 22
- Section 5.5, "Writers to Their Teams and Reviewers," on page 22
- Section 5.6, "Reviewers to the Documentation Team," on page 22

5.1 Project Coordination Calls

As needs arise, Shilpa Bandekar, Shruthy Devendra, Larry Tomkinson, and Catherine Craft will hold conference calls and invite all affected writers to call in as well. Evelyn Sealander and Nicel KM will be copied on all appointments and invited to attend if they want or need to.

Specific call information will be distributed with each GroupWise appointment.

5.2 Documentation Project Manager to the Documentation Team

Shruthy Devendra and Shilpa Bandekar are responsible to communicate the following to documentation team members according to their assigned project management responsibilities:

- Project guidelines and procedures as specified or referenced in this documentation project plan.
- Key milestones and the expectations for each milestone through GroupWise tasks
- Adjustments to the schedule made by the project team
- Other key project information through e-mail
- General status through Core Team and Extended Core Team minutes posted on the OES 2 SP3 project TWiki (https://twiki.innerweb.novell.com/bin/view/OES/OeS2Sp3)

5.3 Documentation Project Manager to the Extended Core Team and Its Members

Larry Tomkinson is responsible to communicate the documentation team's needs and concerns to the extended core team and its members:

5.4 Writers to Documentation Project

See Chapter 7, “Status Reporting Requirements,” on page 25.

5.5 Writers to Their Teams and Reviewers

Writers are responsible to communicate the following to those they work with:

- Any changes to committed work plans
- All required information for each review, such as
  - What is to be reviewed, what to focus the review on, and how/where to access the target documentation
  - How comments should be submitted to the writer

    The standard is to use the User Comments feature.
  - When the review closes
- Submissions of help files and man pages
- Due date for readme items
- Documentation Defects resolved/fixed

  If in Bugzilla, marking the status there is sufficient. If through e-mail or User Comments, a follow-up e-mail should be sent.

5.6 Reviewers to the Documentation Team

See the information in the following sections:

- Chapter 9, “QA Plan,” on page 29
- Chapter 10, “Doc Testing and Feedback Planning,” on page 31
- Chapter 11, “Instructions for Documentation Testers (FAQs),” on page 33
- Chapter 12, “User Comments,” on page 39

6 OES 2 SP3 Documentation Schedule

The documentation schedule is posted and maintained in the Documentation Project Portfolio (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_schedule_view.jsp?pid=19642).

As dates for task completion approach, Shruthy Devendra will communicate with each writer (preferably via e-mail). This will ensure that key dates are communicated and that team members have a consistent and quick way to report task completion status.

7 Status Reporting Requirements

Writers are responsible to communicate the following project information.

- Section 7.1, "Hot Issues," on page 25
- Section 7.2, "Documentation Development Status," on page 25

7.1 Hot Issues

During development team meetings and interactions with team members (testers, engineers, etc.), writers learn of issues that affect them and possibly the documentation team as a whole.

Examples:

- A developer changes an interface after the documentation has been finalized and the help files have been submitted to localization.
- A tester mentions that an unstable component has been pulled from the build and the workaround will probably need to be documented.

Of course, writers are responsible to communicate such things to their managers. They also need to consider whether other team members are affected and should be made aware of the issues. And they should communicate the risks and costs associated with unplanned changes.

It is critical that writers watch for anything that might impact either their ability or the ability of the entire documentation team to meet project commitments. When issues arise, writers should communicate all pertinent details to the appropriate managers and team members as quickly and clearly as possible.

7.2 Documentation Development Status

During the project, all writers need to do the following:

1. Periodically review the documentation schedule (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_schedule_view.jsp?pid=36484) and identify their upcoming tasks and the associated completion dates.

   The schedule lists only high-level tasks. Each high-level task might have two or more subtasks associated with it. For example, for the *Web Doc to novell.com* tasks it is assumed that the content in each affected piece has been updated, passed a technical review, and possibly been through an editing cycle. Writers are responsible for ensuring that all subtasks are completed as well, even though those tasks are not specifically listed on the schedule.
2. Accept GroupWise reminder tasks when they arrive.
3. Plan and complete the work by the dates indicated.
4. When finished with an assignment, mark the associated GroupWise task as completed.

This should always be done before or on the due date.

5 Update the DPP as instructed in the task reminder so that project status reports remain current and accurate by doing the following:

   5a Access the DPP Edit Deliverable Status page (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_deliverable_status_edit.jsp?pid=36484).

   5b Change the review/handoff status for the affected deliverables.

   5c Submit your changes immediately to minimize the risk of overwriting someone else's changes or of having your changes overwritten.

8 Production Plan

The following processes are used to deliver OES 2 SP3 documentation to users:

- Section 8.1, "Engineering Handoffs," on page 27
- Section 8.2, "Publishing on Novell.com," on page 27
- Section 8.3, "On-Demand Publishing," on page 28

8.1 Engineering Handoffs

Writers are responsible to check in all of their files that belong in product builds to their respective engineering build managers.

Each item to be checked in should have a deliverable entry in the DPP, and the check-in status should be recorded there for the milestone for which it was delivered. For more information, see Step 5 on page 26.

8.2 Publishing on Novell.com

Richard Ballard and Arul Kumar Kanniyan coordinate posting the online documentation on the appropriate site (Beta or the Web) for each key milestone.

8.2.1 PDFs Are Always Included

For Beta releases, writers are responsible to create PDF versions of their guides.

For FCS, Rich and Arul create cross-linked PDF files and post them.

8.2.2 Publication Schedule

Shruthy Devendra works with Rich and Arul to determine handoff dates and enter them in the Documentation Project Schedule (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_proj_index.jsp?pid=36484)

Shruthy sends GroupWise tasks to team members to help them track progress and meet their handoff commitments for key project milestones.

8.2.3 Handing Off a Guide for Publication

Writers are responsible to

1. Generate valid XML for each of their guides.
2. Save each guide as PDF.

   Users expect the HTML and PDF versions to match.
3. Commit all changes to the Subversion database.
4. Send an e-mail to Richard Ballard, Shruthy Devendra, and Larry Tomkinson indicating that their guides are ready for publication.

8.3 On-Demand Publishing

Starting with OES 2 SP2 writers can automatically publish guides for review.

8.3.1 The Needs

Writers sometimes need to quickly publish a guide online to conduct an urgent documentation review, resolve a documentation defect, or check whether the HTML presentation is correct for what they are trying to communicate.

Testers rely increasingly on the online documentation for information about the products they are testing. In addition, they are very interested in testing the documentation for completeness and accuracy.

8.3.2 The Solution

To facilitate the needs mentioned above, Richard Ballard and the Documentation Production Team have implemented a process that generates documentation on a regular schedule and publishes it to the Novell.com staging server (wwwtest.provo.novell.com/documentation).

8.3.3 Submitting a Guide for Publication

1. Send an e-mail to Richard or Arul asking that your guide be added to the autobuild list.
2. Generate valid XML for the guide.
3. Save the book as PDF.

   It is critical that the HTML and PDF versions match because testers use the PDF search feature for locating information within a book. (The wwwtest server is not indexed for searching.)
4. Commit the change to the Subversion database

   The guide will be built according to the schedule defined for it.

9 QA Plan

The following standard reviews will help to ensure the quality of the documentation for OES 2 SP3.

- Section 9.1, “Technical Reviews,” on page 29
- Section 9.2, “Peer Reviews,” on page 29
- Section 9.3, “Editing Reviews,” on page 30
- Section 9.4, “Graphics Reviews,” on page 30

9.1 Technical Reviews

Writers should do the following:

1. Request assistance from testing, development, and support engineers to review new documentation content for accuracy and completeness.

   If time permits, engineers can also review other, previously released documentation sections as well.
2. Update the DPP with each deliverable's tech review status.

9.2 Peer Reviews

Writers should do the following:

1. Identify a new chapter or section in one of their guides.
2. Ask the OES 2 SP3 documentation team to review the section for organization, clarity, and usability.
3. Set a deadline for completing the review, and distribute the section for review or identify the access URL.
4. Ask a manager or senior writer to conduct the review session.
5. Schedule and hold a review discussion.
6. Consider the feedback received and make the appropriate changes to the reviewed section.
7. Leverage the review for improving other assigned documentation pieces by extrapolating the feedback to those other pieces.

9.3 Editing Reviews

Writers should do the following:

1. Ensure that at some point in time either Carol or Sachin has edited all the documentation they are assigned.
2. For each deliverable that is changing, schedule an edit with Carol or Sachin as appropriate, according to writer workloads and project milestones.

   To help balance editor workloads, editing should be scheduled as early as it is possible to do with reasonable accuracy. Also, if work is finished early, the documentation pieces should be handed off to the editor as soon as they are finished rather than waiting until the scheduled date.
3. Update the DPP with each deliverable's editing status.

9.4 Graphics Reviews

Writers should do the following:

1. For each deliverable that contains graphics, schedule a Graphics Review with Sandeep to ensure that graphics meet current corporate standards.
2. When the review is complete, update the DPP with each deliverable’s graphics status.

   Mark deliverables with no graphics N/A.

10 Doc Testing and Feedback Planning

This section describes the documentation testing goals and reporting processes to be followed.

- Section 10.1, "About Documentation Testing," on page 31
- Section 10.2, "Scope of OES 3 SP3 Documentation Testing," on page 32

10.1 About Documentation Testing

- Section 10.1.1, "Understanding Doc Testing as Defined for OES 2 SP3," on page 31
- Section 10.1.2, "When Should Testing Be Scheduled?," on page 31
- Section 10.1.3, "Who Can Schedule Doc Tests?," on page 32
- Section 10.1.4, "How Is Feedback Provided?," on page 32

10.1.1 Understanding Doc Testing as Defined for OES 2 SP3

Before planning for OES 2 SP3 documentation testing, be sure you understand the instructions to documentation testers found in Chapter 11, "Instructions for Documentation Testers (FAQs)," on page 33.

10.1.2 When Should Testing Be Scheduled?

Although doc tests can happen at any logical milestones during the project, formally scheduled doc tests should only happen when the following criteria are both met:

- The software is code complete and the UI is at least very stable, preferably frozen.
- The writer confirms that the doc is ready to test.

Any formal testing done before these conditions are met is generally invalid and a wasted effort for all parties involved.

The latest time to start a test cycle is:

- **Active (currently maintained) guides:** Shortly after the final private Beta starts.
- **New or changing guides:** Shortly after Public Beta starts.
- **Post-release guides (such as clustering configuration guides):** No later than two weeks after FCS.

10.1.3 Who Can Schedule Doc Tests?

The documentation project lead can schedule a documentation test for the entire project with the testing manager and writing team.

Alternatively, a writer can coordinate the testing of an individual section or guide with the testing manager.

As arranged with the testing manager, writers tell the manager or team when the section is ready to test, where the target documentation is located, and the date by which the test should be completed so that work on the document can continue.

10.1.4 How Is Feedback Provided?

Doc testers provide feedback according to the media (paper, online, etc.) tested, as explained in Section 11.4, "What Are the Feedback Mechanisms for Each Documentation Type?," on page 35.

10.2 Scope of OES 3 SP3 Documentation Testing

The OES 2 Documentation Team will see that all guides and setions that haven't been doc tested are doc tested for the OES 2 SP3 release. This includes, as a minimum, the guides for the components shown in the following table.

Guide/Component	Author	Date
For SP3, doc testing was scheduled before this information was included. The section is retained, however, for use in future projects, specifically Ponderosa.		

11 Instructions for Documentation Testers (FAQs)

Your feedback is critical because it enables the writing team to align every doc detail with the OES 2 SP3 software and to support user tasks with documentation that is accurate and usable.

This section answers the following questions:

- Section 11.1, "What Is Documentation Testing in OES 2 SP3?," on page 33
- Section 11.2, "When Do I Submit Feedback?," on page 34
- Section 11.3, "What Pieces Should Be Tested?," on page 34
- Section 11.4, "What Are the Feedback Mechanisms for Each Documentation Type?," on page 35
- Section 11.5, "Why Not Use Bugzilla for HTML Feedback?," on page 36
- Section 11.6, "What Should I Report?," on page 36
- Section 11.7, "Are There Things I Should Not Report?," on page 37
- Section 11.8, "How Do I Report Missing ECO and New Feature Items?," on page 37
- Section 11.9, "What About Issues That Originate in Software Bugs?," on page 37
- Section 11.10, "How Do I Check the Status of My User Comments?," on page 38
- Section 11.11, "What Do the User Comment Severity Levels Mean?," on page 38

11.1 What Is Documentation Testing in OES 2 SP3?

- Section 11.1.1, "What OES Doc Testing Is," on page 33
- Section 11.1.2, "What OES Doc Testing Is Not," on page 34

11.1.1 What OES Doc Testing Is

Documentation testing in OES involves using the software as described in the documentation, step-by-step. It includes evaluating and improving the user experience.

For example, testing an installation procedure involves four steps:

1. Checking that any required prerequisites are clearly identified and explained.
2. Verifying that the instructions for starting the installation process are clear and work as described.
3. Verifying that all of the steps involved in the installation process are documented and accurate, that menus and dialog boxes are correctly named, that required inputs are clearly identified, and that all program outputs are accurately described and explained.

4. Verifying the following:
   - That completing the documentation instructions results in a completed installation.
   - That no "loose ends" remain
   - That all of the "next steps" are clearly set forth and linked.

11.1.2 What OES Doc Testing Is Not

OES documentation testers must not try to conduct editorial or technical reviews.

- **Editorial Reviews:** Doc testing is not an editorial review.

  Testers are not required to be familiar with Novell style guides and are usually not qualified to comment on the language aspects of documentation.

  Novell style is based on American English. What sounds good and "natural" to a documentation tester in one non-US Novell location is usually only personal preference and often makes the documentation sound stilted, unnatural, and confusing to users in other areas of the world.

  For information on editorial reviews, see Section 9.3, "Editing Reviews," on page 30.

- **Technical Reviews:** Doc testing is not the same as a technical review that involves only reading the sections and then commenting on technical accuracy issues.

  Of course, doc testers should read the instructions and point out any technical inaccuracies, but such comments should be only incidental and must not be the primary focus of a doc test. For information on technical reviews, see Section 9.1, "Technical Reviews," on page 29.

11.2 When Do I Submit Feedback?

Whenever you spot a problem, please report it. Formal doc tests are great, but feedback is needed and welcome at any time.

11.3 What Pieces Should Be Tested?

- Section 11.3.1, "Focus on the HTML Guides," on page 34
- Section 11.3.2, "Don't Emphasize Context-Sensitive Help," on page 34
- Section 11.3.3, "Ignore Sections Under Construction," on page 35

11.3.1 Focus on the HTML Guides

You should test the HTML guides posted on the Internal Test Server (http://wwwtest.provo.novell.com/documentation/oes2).

11.3.2 Don't Emphasize Context-Sensitive Help

You are welcome to look at the context-sensitive help during a doc test, and you can provide feedback on the help anytime, context-sensitive help doesn't usually include step-by-step instructions, and it is therefore not the primary focus of doc testing. Rather, context-sensitive help is usually checked as part of a technical review of the help itself.

11.3.3 Ignore Sections Under Construction

You should not comment on HTML pages that display the "Under Construction" graphic unless the doc is supposed to be in a final state.

The Documentation Under Construction graphic is a clear indication that the writer is working to fix inaccuracies on the page.

11.4 What Are the Feedback Mechanisms for Each Documentation Type?

Please use the following reporting methods:

Doc Type	Method
HTML Doc	**User Comments:** Use this method for all doc feedback regarding individual HTML pages. 1. Check the *User Comments* section at the bottom of the HTML page in question to ensure that your comment doesn't duplicate something that was already submitted. 2. Click *Add Comment* . 3. Type your comment and e-mail address. For example, joeuser@novell.com. 4. Click *Submit*. The writer of the guide will e-mail you if clarification is needed. The writer will also e-mail you about the resolution of the comment (fixed, posted, enhancement request logged, etc.)
Missing Doc	◆ **E-mail:** If you're not sure that the doc is scheduled for availability, send an e-mail to the writer. ◆ **Bugzilla:** If you know that the doc is scheduled for availability but is not there, log a bug.
Paper	If the writer distributed hard copy documentation for review, do one of the following: ◆ **Hard copy markup:** If you prefer, mark up your copy and then return it to the writer. A face-to-face review of your notes is generally helpful but not required. ◆ **E-mail:** This also works well. Be sure to specify the section and paragraph where the problem is found in the guide.
PDF Doc	If the writer asked for a review of a PDF file, do one of the following: ◆ **PDF Notes:** These work well if the reporter has a version of Acrobat that supports them. ◆ **E-mail:** This also works well, but be sure to specify the section and paragraph where the problem is found in the guide.

Doc Type	Method
UI Help	◆ **E-mail:** This is sufficient and preferred for most help issues. ◆ **Bugzilla:** Use this if: ◆ The problematic help has been packaged or published for final release. ◆ You don't know who the component writer is. ◆ A bug is assigned to a software defect that will be addressed in the future, but a doc “patch” is needed in the interim. ◆ A bug is assigned to a software defect that will not be addressed in the future and a doc “patch” is needed.

11.5 Why Not Use Bugzilla for HTML Feedback?

Bugzilla overhead can’t be justified for most doc issues. For example, a single spelling error defect generates a handful of e-mails that multiple people must all take time to deal with. 10 spelling errors multiplies the effect by 10.

More importantly, doc defects tend to increase toward the end of development cycles. This can distort and skew critical trending and other metrics, thus generating unnecessary stress for project managers and documentation team members alike.

Some doc issues are not fixed until after FCS. For example, if a new section is needed and there isn’t time to get it written before FCS, it becomes an “Escape” issue in Bugzilla that negatively impacts IPD product quality metrics.

11.6 What Should I Report?

You should report the following things:

- Confusing wording.

  IMPORTANT: Be sure to explain why the documentation section confuses you. Do not, however, rewrite the sentence or section for the writer (see “Editorial Reviews:” on page 34).

- Incorrect or missing information.

  This includes but is not limited to the following:

  - Missing or incorrect prerequisites
  - Missing or incorrect procedures and/or steps within a procedure
  - Screen shots that display old or obsolete interfaces
  - Inaccurate command syntax, etc.

- Security issues, such as the use of real routable IP addresses and real Novell employee usernames and e-mail addresses.

11.7 Are There Things I Should Not Report?

Yes. Do not report the following things:

- Grammatical errors
- Misspellings
- Language preferences

  Doc testers are not tasked with doing a language edit (see "Editorial Reviews:" on page 34).

The project editor will catch these at the end of the development cycle.

11.8 How Do I Report Missing ECO and New Feature Items?

New feature requests and ECOs flow down from the Engineering Requirements in FATE (or from Product Requirements Documents), not Bugzilla. As active development team members, writers should know about these features and deliver them as part of the new or updated documentation.

Typically, these are approved by Product Managers and Engineering Managers as part of the formal Documentation Plan.

If you are testing a document that the writer has indicated is complete, and you can't find information on a new feature or ECO, use the following procedure:

1. If you know who the responsible writer is, contact him or her using e-mail.
2. If you can't determine who the responsible writer is, log a defect against the documentation, indicating what you are seeking and where you have looked. The defect will be assigned to the appropriate writer.

11.9 What About Issues That Originate in Software Bugs?

Bugzilla is the primary location for defects found in the administrator and user help interfaces when testing the product itself. Documentation also relies on Bugzilla for documentation requests that result from engineering defects and fixes for those defects, such as the following:

- New panels or dialog boxes in an existing interface
- Changes to the interface
- Changes to how the product works
- Issues that won't be fixed and need to be documented as caveats
- Issues that need a Readme item

Bugzilla-based documentation assignments should flow naturally from existing engineering bugs. Simply assign the defects to the writers. When the documentation is updated, the writer will reassign it to the last owner for verification that the documentation has been changed for that defect. It will be up to the engineer to verify the documentation and mark the bug fixed.

11.10 How Do I Check the Status of My User Comments?

To check the status of comments you have logged, do the following:

1. Browse to the User Comments Database (http://doccomments.provo.novell.com/admin/main?/).
2. In the left navigation bar, click *Search*.
3. (Conditional) If you want to filter your search, select from the lists of available options. For example, select the entries for OES 2 in the *Product* column.
4. In the *Customer E-mail* field, type your novell e-mail name without @novell.com. For example, type `joeuser`, not `joeuser@novell.com`.

   You can type as many e-mail names as you want, separated by spaces.

   You can also type `novell.com` to select all submissions by Novell employees.
5. Click *Search*.

11.11 What Do the User Comment Severity Levels Mean?

The following definitions are used in the Doc Comments database. Keep in mind that these apply to documentation components, *not the software they document*.

- **Blocker:** Key documentation is missing for widely accessed public release or critical testing and review. The documentation cannot be published until this issue is resolved.
- **Critical:** Documentation prescribes or doesn't warn against actions that cause data loss, corruption, or destruction.
- **Major:** Documentation is missing information, provides inaccurate information, or contradicts itself and successful task completion is unlikely. Documentation that is strategic, frequently accessed, or very visible is missing information or provides inaccurate information. Successful task completion might not be possible.
- **Normal:** Documentation that is less frequently accessed and less visible is missing information, or provides inaccurate information, but successful task completion is probable.
- **Minor:** Documentation contains stylistic or formatting issues, but functionality is not hindered.
- **Enhancement :** Suggestion for improvement.

NOTE: Blocker defects are fixed immediately. Critical and major defects are addressed by Public Beta. All user comments are addressed by FCS, with some exceptions being published by FCS+30 days.

---

12 User Comments

The Documentation organization has an existing work process flow for dealing with User Comments in the normal course of business. Incoming comments should be triaged and included in workload plans. This section outlines additional guidelines for handling user comments within the OES 2 SP3 project.

- Section 12.1, "Preparing for User Comment Submission," on page 39
- Section 12.2, "OES User Comment Goals," on page 39
- Section 12.3, "Evaluating Suggestions for Rewrites," on page 40
- Section 12.4, "Handling User Comments," on page 40

12.1 Preparing for User Comment Submission

As an OES writer, you are responsible for making sure your guides are posted on the Novell Test Server (wwwtest.provo.novell.com) via the doc autobuild process.

1. Submit a production request in Bugzilla for each of your guides to be autobuilt on the wwwtest server.
2. Update the book in Frame and export it as XML.
3. Commit the saved XML to Subversion.
4. Verify that the autobuild process is working by checking the guide on the wwwtest server.

   It can take up to two hours for the changes to synchronize. Check with your production specialist if you don't see your changes after two hours.
5. Repeat from Step 2 each time you need the guide to be published.

User Comments on the wwwtest server are automatically displayed at the bottom of the affected online page. You are automatically notified of new comments via e-mail.

12.2 OES User Comment Goals

There are two general goals for OES 2 SP3:

- Address all Critical and Major user comments from testers and engineers by Public Beta.
- Address all user comments by FCS.

The Documentation Project Manager can approve exceptions for publication by FCS+30 days if the required information is not available or other factors outside of doc team control prevent publication.

12.3 Evaluating Suggestions for Rewrites

You should always carefully evaluate user comments that indicate that a section is confusing, especially when the commenters explain why they are confused.

On the other hand, you should generally ignore all attempts by internal Novell testers to rewrite your documentation. In most cases, these kinds of suggestion are not helpful. (Documentation testers are specifically instructed to not submit rewrites. See "Editorial Reviews:" on page 34 and "Are There Things I Should Not Report?" on page 37.)

If a suggested change seems to be an improvement, submit it to an editor before incorporating the change.

12.4 Handling User Comments

1. Evaluate the user comment and make the changes in your documentation.

   Make sure you also consider any ripple effects, such as other sections that might require changing or links that might need to be created or removed.

2. Publish the change on the wwwtest server by exporting the book to XML and committing the change to Subversion.

3. After corrections are published on the test server, mark the comments *Fixed*.

   IMPORTANT: Marking comments as *Fixed* removes them from the online page, so make sure the correction is published before doing this.

4. Notify each submitter via e-mail that the updated doc is available for verification.

   Be sure to include the original comment and a link to the HTML page in the e-mail. An easy way to do this is to forward the original User Comment notification to the submitter with the notification.

13 Localization Plan

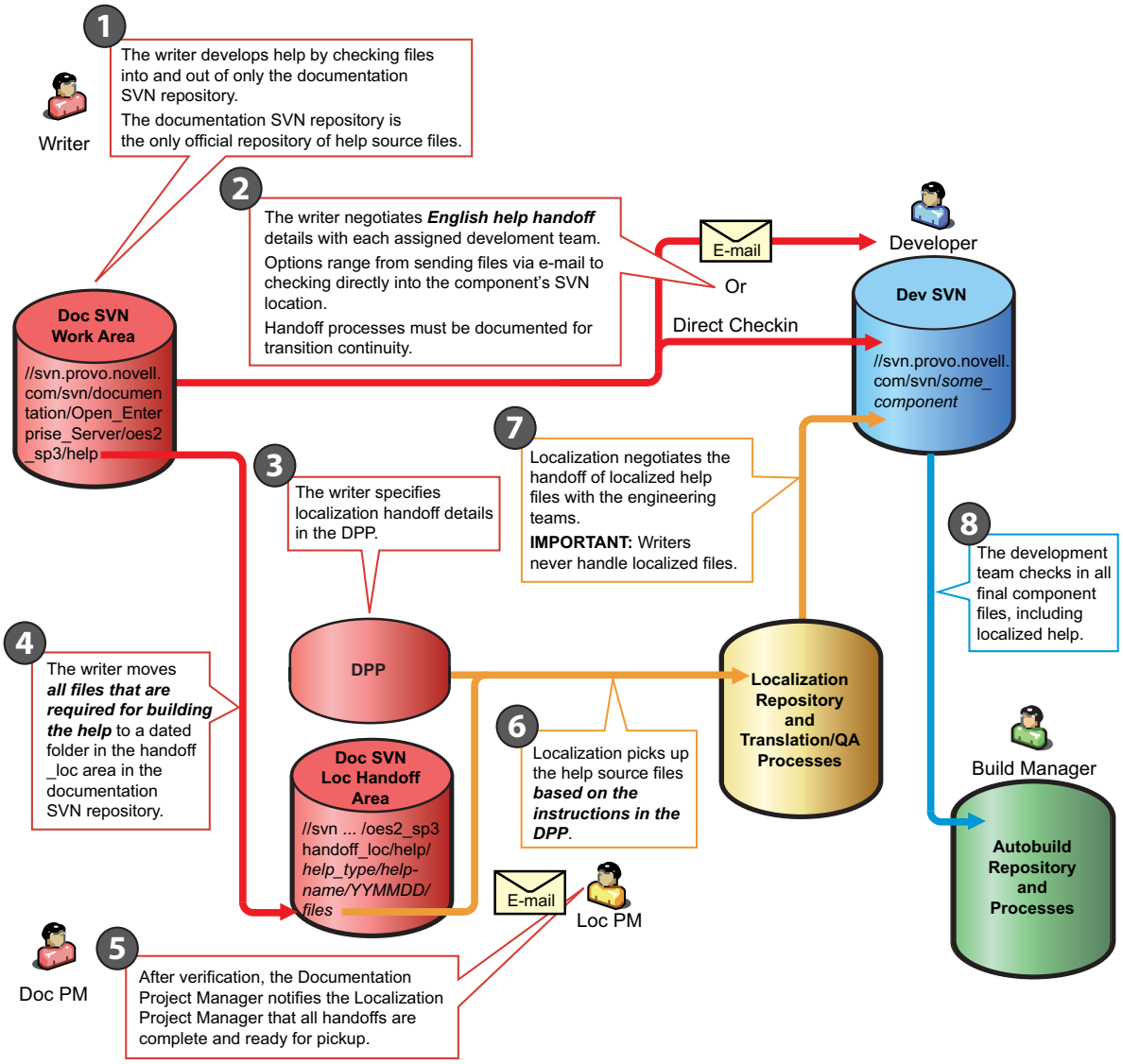
Localization scope is set by the Localization Project Manager, Padraig Dillon, and recorded in the DPP (https://et.innerweb.novell.com/project_pages/uxs/doc/doc_proj_index.jsp?pid=19642). The OES 2 SP3 Documentation Project Lead works with Padraig to determine handoff dates for deliverables and adds the information to the DPP.

Writers hand off their deliverables using the DPP. The doc project lead coordinates all handoffs and is the point of contact with localization for any deviations from the plan recorded in the DPP.

13.1 Overview of Help Localization Processes

The following graphic illustrates the standard help development and localization process as agreed between the localization, development and documentation groups.

Figure 13-1 *OES Help Development and Localization*



13.2 Keeping Localization Costs As Low As Possible

Today's economic climate demands that localization costs be kept as low as possible and Novell has developed the the processes outlined in Figure 13-1 on page 42 to ensure that this happens.

The following summarizes the key points of Novell's localization processes for all help files.

- Doc SVN and Dev SVN serve different purposes and must be kept separate from each other.
  - There must be a standard location for all help source files for continuity as team members come and go.
  - The files that deliver English help to end users are often not the source files from which the help is built. One prime example of this is a help handoff that is built from a Frame-based, profiled guide. HTML is checked into the build, but producing that HTML requires both the XML source from Frame and the profile used to build the help.

- Documentation Production personnel have heavy workloads. Assisting with help files is outside of their assigned scope, and requests for them to get involved with help are not appropriate.

  This includes such activities as helping to automate the transfer of files from the Doc SVN to a Dev SVN.

- Writers also have heavy workloads and are prohibited from direct involvement with localized files.

  Although this sounds harsh, deviation from this policy almost always results in increased localization costs.

  Only localization personnel can properly assess the impact of issues that arise. For example, not all changes in English files that were previously submitted, such as typos or formatting glitches, require retranslation.

- Localization cost-savings have dictated many of the processes outlined in this section.

  Padraig Dillon has emphasized repeatedly that handoffs must occur in the standard way because deviations are expensive.

A Working with Subversion

This section is adapted from the instructions and information on the Innerweb Wiki—Doc Team Subversion Implementation - FY08 TWiki (https://wiki.innerweb.novell.com/index.php/Doc_Team_Subversion_Implementation_-_FY08) to accommodate OES-specific guidelines and issues. All OES team members should also be familiar with the information on the TWiki.

Subversion, the version control standard for the Provo and Bangalore sites, augments the docsys/FrameMaker toolset that Novell writers use.

Source control standardizes work practices for all writers as follows:

- Frame guides and help source files are stored in the Subversion repository, and you never access them directly.
- Working copies of the Frame guide and help files are created and maintained on either a local hard drive or in a home directory on a server.

  A Subversion client (TortoiseSVN) that you install on your workstation, synchronizes the working copies with the main Subversion repository.

Read and follow the instructions in the following sections.

- Section A.1, "Gaining Access to the Repository," on page 45
- Section A.2, "Installing the TortoiseSVN Client," on page 46
- Section A.3, "Creating a Work Directory on Your Local Drive or Home Directory," on page 46
- Section A.4, "Understanding Important TortoiseSVN Menu Options," on page 47
- Section A.5, "Populating Your Work Directory," on page 49
- Section A.6, "product.ini FAQ," on page 52
- Section A.7, "About the Standardized product.ini File," on page 54
- Section A.8, "Always Use TortoiseSVN Options in Windows Explorer," on page 54
- Section A.9, "Accessing the TortoiseSVN Documentation," on page 54

A.1 Gaining Access to the Repository

Because all Frame source files are stored in a Subversion repository, you need access to the repository. To gain access, you first need an account on Novell InnerForge, then Richard Ballard or Arul Kumar Kanniyan can grant you access.

1. If you know you have an account on InnerForge, skip to Step 4. Otherwise, continue with the next step.
2. Log into the Innerweb and access the Novell InnerForge Main Page (https://forge.provo.novell.com/modules/news/)

3 If your name and an option to log out appear in the upper-left corner of the page, that means you are already registered using your Innerweb (eDirectory) username and password. If not, click the *Register with InnerForge* button and complete the registration process.

4 Log a Production Request bug for Rich and Arul, tell them you are registered with InnerForge, and ask them to grant you access to the repository.

A.2 Installing the TortoiseSVN Client

To use Subversion with Frame, you must install the TortoiseSVN Subversion client for Windows.

Installing the client adds Subversion menu options to both your Windows Explorer *File* and right-click menus.

To install the client, do the following:

1 In your Web browser, access the TortoiseSVN Web site (http://tortoisesvn.net/downloads). http://tortoisesvn.net/downloads, then install the client.

2 Download the 32-bit Installer.

3 Open and run the installer on your workstation, accepting the license, etc. as required.

4 When you are prompted to restart, close any applications that you have open and restart your computer.

A.3 Creating a Work Directory on Your Local Drive or Home Directory

Now you need a work directory. This is where you will download working copies of files from the Subversion repository and then open, create, and change the files with FrameMaker and applicable help authoring tools.

1 Choose a drive with enough free disk space for your files and all the guides you will need to link to.

For example, if your guides require 100 MB of disk space and you must link to other guides that consume 200 MB of disk space, you will need at least 300 MB of free disk space on the drive, and you might want to plan for 400 to 500 MB to accommodate future growth.

However, if your disk is adequate but not ample, this is not a cause for stress. Your work directory can be replaced in the future if the disk gets too full, or the drive fails, etc. The ability to restore your work area at any time is one of the main advantages of having a version control system.

2 On the drive you have identified, create a work directory named `doc_projects`. This is sometimes referred to as the *Trunk* directory.

You can actually specify other names for the directories outlined in these instructions, but using the directory names given here will help you with creating the required product.ini file and with understanding and using future instructions. It will also make it easier to navigate the repository structure if you use the same names that it uses.

3 In the `doc_projects` directory, create a nested subdirectory structure as follows:

```
doc_projects\Open_Enterprise_Server\oes2_SP3\frame
doc_projects\Open_Enterprise_Server\oes2_SP3\help
```

IMPORTANT: Because the `product.ini` file is shared by all team members, it is critical that all instructions regarding the directory structure you create for your local repository be followed exactly.

A.4 Understanding Important TortoiseSVN Menu Options

Working with the Subversion repository requires that you use some key TortoiseSVN menu options:

- Section A.4.1, "Add," on page 47
- Section A.4.2, "Checkout," on page 47
- Section A.4.3, "Delete," on page 49
- Section A.4.4, "SVN Commit," on page 49
- Section A.4.5, "SVN Update," on page 49

A.4.1 Add

New files that you create in your work directory are not automatically added to Subversion. For example, you can add a chapter to a book in Frame 9 just as you always have, and the book will generate and save just fine, but until you use the *TortoiseSVN > Add* menu option, the new chapter is not tracked by Subversion.

To add a file to Subversion, do the following:

1. In Windows Explorer in the book directory, right-click the file and select *TortoiseSVN > Add*.
2. Click *OK*.
3. Right-click the file again and select *SVN Commit*.
4. Click *OK > OK*.

   A green check mark is added to the file icon, indicating that the file is now being tracked by Subversion.

A.4.2 Checkout

You use the Checkout option to initially download the books you need from the Subversion repository to a given work area. However, even though you probably won't use it very often, it is vital that you understand how Checkout works.

Checkout Copies Children Only

The Checkout operation copies all of the children (subdirectories and files) from a source directory (the *URL of repository* field in the Checkout dialog box) to a target directory (the *Checkout directory* in the Checkout dialog box).

This can be a bit confusing to those who expect that the source directory itself will also be copied.

An easy way to make sure a checkout operation goes as expected is to make sure that the directory names for both the source and the target match. For example, if you are copying all of the books for a project, then the *URL of repository* and the *Checkout directory* fields should both end with `frame`. All of

the book subdirectories will then be copied to the target. On the other hand, to copy just one book, both fields should end with the name of the book directory. To copy a set of help files, both fields should show the same starting point in the help directory structure

Where the Checkout Option is Available

The Checkout option is only available on directories that aren't being tracked by the TortoiseSVN client, as indicated by a modified icon in Windows Explorer.

For example, after you've created a `frame` directory and checked out its books from the repository, the icon for the directory changes in Windows Explorer and you can't access the Checkout option for the `frame` directory again. However, any new books, chapters, etc. that are added to the directory in the repository by other writers should be downloaded the next time you run an SVN Update operation.

Only Check Out a Project, a Book, or a Complete Set of Help Files

Although you can check out a single file, in actual practice you should check out either a project, a book within a project, or the help files for an application or plugin, etc.

If you check out a book, you get all the files and subdirectories that are children of that book in the repository. If you check out a project, you get all of the books that are children of that project in the repository. If you check out a set of help files, you get all the files and subdirectories that are included in the set.

This has implications for checking out large projects, such as OES 2 SP3. For example, the OES 2 SP3 > frame directory on the Hain server, uses almost 4 GB of disk space. Although this will hopefully be reduced as writers clean up their files, downloading and updating such a large set of files is probably not reasonable for most writers.

Unless you need to link to the majority of the documentation set, you will want to check out the books you need one at a time.

Not All Files Belong in the Repository

Not all files that are in a project or book are stored in the repository. Generally, the files in the repository are those that are either needed by FrameMaker or the help authoring tool, or files that are required for generating a doc deliverable.

Therefore, Frame, graphics, XML, PDF, and ID files are stored in the repository, while writer notes and project related documents are not stored. However, they can be added by the writer for either backup or version control purposes.

Some files that exist in the work directory should not be stored in the repository. For example, the `product.ini` file in the frame directory could potentially be different on each machine, and the `product.ini` file that one writer uses could break the environment on another writer's machine if it were to be downloaded from the repository.

A.4.3 Delete

If you no longer need a file that is tracked by Subversion, you must delete it using the *TortoiseSVN > Delete* option. Otherwise, the next time you run an SVN Update, the file will reappear. Be aware that Delete only means the file is removed from your work directory and is no longer tracked by Subversion. Older versions of the file that were tracked by Subversion are still available for retrieval.

A.4.4 SVN Commit

After you have checked out a book and made changes to it, you use this option to send your changes to the repository for synchronization.

A.4.5 SVN Update

This option checks the repository for changes that are needed to the work directory. This is used when

- You want to update local copies of books that you don't own but you link to.
- You have multiple work areas. For example, one at home and one in the office, and you want to download the updates you have made in the other work area.

  IMPORTANT: This requires that all changes in the other area have been committed and all file locks have been released.

- You share a book or set of help files with other writers and you need to download the updates that they have made.

A.5 Populating Your Work Directory

Now it's time to check out (download) working copies of your guides and help files that you will need. Complete the steps in the section that applies to your situation.

- Section A.5.1, "Checking Out Only the OES 2 Guides You Need," on page 49
- Section A.5.2, "Checking Out the Help Files You Need," on page 50
- Section A.5.3, "Checking Out the Entire OES 2 SP3 Project," on page 50
- Section A.5.4, "Checking Out Other Products you link to," on page 51

A.5.1 Checking Out Only the OES 2 Guides You Need

As mentioned in the previous section, most writers only need access to a small set of books. Complete the following steps.

1. In Windows Explorer, right-click the `frame` folder you created in Step 3 on page 46 and select *SVN Checkout*.
2. (First time only) In the *URL of Repository* field, type

   https://svn.innerweb.novell.com/svn/documentation
3. Click the browse button (with three dots) to the right of the *URL of Repository* field.

4 If prompted, type your InnerForge username and password and select the *Save Authentication* option (unless you share your workstation with another writer), then click *OK*.

   Saving your authentication information eliminates the need to enter your username and password each time you interact with the repository. In most cases this will work best. However, if you share your workstation with another writer, you should both authenticate separately.

5 Navigate to and select the `Open_Enterprise_Server > oes2_SP3 > frame > bookname` folder, and then click *OK*.

6 In the *Checkout directory* field append a back slash (\) and the exact *bookname* as listed in the repository (in Step 5), then click *OK*.

7 Click *Yes* to confirm that you want the bookname directory created.

8 Repeat these steps for all of the books you own and for those you need to link to.

A.5.2 Checking Out the Help Files You Need

In almost all cases, you only need to download the help files you own.

Complete the following steps.

1 In Windows Explorer, right-click the `help` folder you created in Step 3 on page 46 and select *SVN Checkout*.

2 (First time only) In the *URL of Repository* field, type

   https://svn.innerweb.novell.com/svn/documentation

3 Click the browse button (with three dots) to the right of the *URL of Repository* field.

4 If prompted, type your InnerForge username and password and select the *Save Authentication* option (unless you share your workstation with another writer), then click *OK*.

   Saving your authentication information eliminates the need to enter your username and password each time you interact with the repository. In most cases this will work best. However, if you share your workstation with another writer, you should both authenticate separately.

5 Navigate to and select the `Open_Enterprise_Server > oes2_SP3 > help > help_file_set_name` folder, and then click *OK*.

6 In the *Checkout directory* field append a back slash (\) and the exact *help_file_set_name* as listed in the repository (in Step 5), then click *OK*.

7 Click *Yes* to confirm that you want the directory created.

8 Repeat these steps for all of the help file sets that you own.

A.5.3 Checking Out the Entire OES 2 SP3 Project

As mentioned in the previous section, to keep things efficient and simple, you can check out all a project's books in a single operation. Complete the following steps.

1 In Windows Explorer, right-click the `frame` folder you created in Step 3 on page 46 and select *SVN Checkout*.

2 In the *URL of Repository* field, type

   https://svn.innerweb.novell.com/svn/documentation/Open_Enterprise_Server/oes2_SP3/frame

3 Click the browse button (with three dots) to the right of the field.

4 If prompted, type your InnerForge username and password and select the *Save Authentication* option (unless you share your workstation with another writer), then click *OK*.

   Saving your authentication information eliminates the need to enter your username and password each time you interact with the repository. In most cases this will work best. However, if you share your workstation with another writer, you should both authenticate separately.

5 Click *OK*.

6 Click *OK* to check out (download) all the OES 2 SP3 books in the repository.

A.5.4 Checking Out Other Products you link to

If you link to books in bundled products, such as eDirectory, you will need to have copies of the Frame books you link to in your repository as well. Complete the following steps.

1 Open the SVN Repository Browser and find the project guides that you need to link to.

2 Using the repository directory structure as a model, create the same structure down to the guide level in your `doc_projects` directory.

   For example, if you need to link to the eDirectory 8.8 guides, create the following directory structure:

   `doc_projects\edirectory\edir88\frame`

3 In Windows Explorer, right-click the `frame` folder you just created and select *SVN Checkout*.

4 Click the browse button (with three dots) to the right of the *URL of Repository* field.

5 If prompted, type your InnerForge username and password, then click *OK*.

6 If you want to check out all of the bundled product's books, do the following:

   6a Browse to the parent directory of the product guides you need to link to.

      The directory should be named either `frame` or `manuals`, depending on how the product team has chosen to define their directory structure.

   6b Click *OK*.

   6c Click *OK* to check out (download) all the books in the repository.

7 To check out individual books within the bundled project, do the following:

   7a Navigate to and select the *`bookname`* folder for the book you want to link to, then click *OK*.

   7b In the *Checkout directory* field append a back slash (\) and the exact *bookname* as listed in the repository (in Step 5), then click *OK*.

   7c Click *Yes* to confirm that you want the bookname directory created.

   7d Repeat these steps for all of the books you need to link to.

8 If you checked out the entire book parent directory, the `product.ini` file for the bundled product should have been checked out as well. Verify that your work directory for the bundled product contains a `product.ini` file. If it does, you are done. If it is doesn't, continue with Step 9.

9 If you checked out individual books or a bundled product's book parent directory didn't include a product.ini file, do the following:

   9a Open the SVN Repository Browser and navigate to the bundled product's book parent directory.

   9b If a `product.ini` file is listed, open it and copy its contents to your workstation's clipboard.

9c In your local working area for the bundled product, open a new file using Notepad, copy the contents of your clipboard to the file, and save the file as `product.ini`.

9d If a product.ini file is not listed in the SVN Repository Browser, contact Richard Ballard and request a `product.ini` file for the bundled product. Then put the file that Richard sends in the book parent directory of the bundled product.

A.6 product.ini FAQ

If you want to learn more about how `product.ini` files work, read the following FAQs.

- Section A.6.1, "What is the function of the product.ini file?," on page 52
- Section A.6.2, "Where is the product.ini file located?," on page 52
- Section A.6.3, "Is a product.ini file required in each book parent directory?," on page 52
- Section A.6.4, "I only link to sections within my own book. What do I need?," on page 52
- Section A.6.5, "What if I link to other books in the OES 2 doc set?," on page 53
- Section A.6.6, "Why must the idfiles directory be included in each Frame book?," on page 53
- Section A.6.7, "I have links to books in other products. What do I need to do to link to these other products?," on page 53

A.6.1 What is the function of the product.ini file?

The `product.ini` file provides the NovDoc tools with a product ID for each book in a given product. This ID is issued by the production team, and it must be listed in the Conformance attribute of each book file for valid XML/HTML generation. The ID for all Open Enterprise Server 2 releases, including support pack releases, is `oes2`.

The `product.ini` file also lists the product IDs of products that are logical link targets, such as bundled products, and it provides the path to each target product's book parent directory. Because it is the platform for many other Novell products, the OES 2 doc TOC links to the books in roughly 24 other Novell products.

A.6.2 Where is the product.ini file located?

In the book parent directory of each product, which is usually named `frame` or `manuals`.

A.6.3 Is a product.ini file required in each book parent directory?

Yes, because both FrameMaker linking and generating XML/HTML with the NovDoc tools require it.

A.6.4 I only link to sections within my own book. What do I need?

Your `product.ini` file must contain a [product] section that specifies the product ID. The section is formatted as follows:

```
[Product]

Name=product_ID
```

NOTE: The `[Products we can link to]` section, although present in most product.ini files, is not required unless you are linking to books for other products.

A.6.5 What if I link to other books in the OES 2 doc set?

In addition to the `product.ini` file mentioned above, you will also need the following:

- An SVN download in your `doc_projects` (or equivalent) work directory of each OES 2 book that you want to link to. See "Checking Out Only the OES 2 Guides You Need" on page 49.
- An `idfiles` subdirectory in each book directory, that contains XML files—one for each frame file in the book (except the TOC file of course). These files are regenerated each time you update a book.

A.6.6 Why must the idfiles directory be included in each Frame book?

Its XML files provide the link target identifiers and the title strings that the NovDoc tools use for creating XRefInt, XRefIntText, XRefExt, and XRefExtText links in Frame.

A.6.7 I have links to books in other products. What do I need to do to link to these other products?

If you are following the instructions in this doc plan and using the standardized product.ini file (see About the Standardized product.ini File (page 54), your environment is configured for bundled product links.

If you have a different working environment, you must do two things:

1. Perform an SVN download all of the bundled books you want to link to. Follow the instructions in "Checking Out Other Products you link to" on page 51.
2. Make sure that the `[Products we can link to]` section in your `product.ini` file contains a line for each bundled product you have downloaded. Refer to the standardized OES 2 product.ini for an example.

   Each bundled product line must contain

   - The product ID that is shown as the product name in the bundled product's `product.ini` file. For example, for eDirectory 8.8 the product ID is `edir88`.
   - An Equals sign (=).
   - The direct or relative path to the bundled product's book parent directory.

   For example, the line for your eDirectory 8.8 download might contain

   ```
   edir88=D:\doc_projects\edirectory\edir88\frame
   ```

   or

   ```
   edir88=..\..\..\edirectory\edir88\frame
   ```

A.7 About the Standardized product.ini File

Because of generating and linking problems encountered during the initial phase of the SVN rollout for OES 2 SP3, Larry has developed a standardized `product.ini` file that you can use if you have implemented the directory structure outlined in Section A.3, "Creating a Work Directory on Your Local Drive or Home Directory," on page 46 and the sections that follow it.

The `product.ini` file is included in the `oes2_SP3\frame` directory in Subversion, but you will only have it in your work directory if you have SVN downloaded the entire OES 2 doc set. If you are not comfortable with retrieving the file from Subversion for your work directory, contact Larry, and he will send it to you in an e-mail.

A.8 Always Use TortoiseSVN Options in Windows Explorer

The TortoiseSVN menu options are required for all interactions with the Subversion repository. They also replace some of the actions that you would normally use native Windows Explorer functionality to accomplish, such as renaming, moving, or deleting directories and files. The Add and Delete options are especially critical and are explained in Section A.4, "Understanding Important TortoiseSVN Menu Options," on page 47.

Failure to comply with this directive will cause the work directory to get out of sync with the repository.

A.9 Accessing the TortoiseSVN Documentation

After the client is installed, you can easily access the documentation by right-clicking any directory in Windows Explorer and clicking *TortoiseSVN > Help*.

B Maintaining the OES Online TOC Pages

One of the key elements of the OES 2 documentation set is its rather extensive online TOC.

The information in this section is largely for reference by the OES information architect and the documentation project manager. However, all team members should be familiar with the subject matter so that they can avoid causing broken links and assist with TOC maintenance for their assigned services and components.

- Section B.1, "Overview," on page 55
- Section B.2, "Working in the TOC Book," on page 62
- Section B.3, "Producing the TOC," on page 70
- Section B.4, "The OES TOC Is Also an Important Tool," on page 72
- Section B.5, "Looking Forward to Ponderosa," on page 73

B.1 Overview

The OES online TOC was developed to provide direct access to information and instructions using industry-standard and user-requested terminology. In many ways it is like a multi-subject index of tasks and information, but it serves the same basic function as the other Novell documentation online TOCs do for their respective products—it is the main access portal to OES product documentation.

- Section B.1.1, "Historical Background," on page 55
- Section B.1.2, "The Current OES 2 TOC," on page 57
- Section B.1.3, "About the TOC Template," on page 61
- Section B.1.4, "About the TOC Macros," on page 62

B.1.1 Historical Background

- "The Traditional NetWare TOC" on page 56
- "A New Challenge" on page 56
- "OES 1 TOC: Limited Success" on page 56
- "A Paradigm Shift" on page 57

The Traditional NetWare TOC

Providing logical access to information has always been a priority for the Novell documentation group.

Prior to OES, Novell traditionally offered NetWare customers two things in its online TOC:

- An A-Z list of the guides
- An alternative approach that grouped related guides together under topics, such as Welcome to NetWare, Installing, Securing and Controlling, various Managing topics, Optimizing, Troubleshooting, etc.

To find task information, one opened a guide that looked promising and then navigated through its table of contents looking for the information. Of course the terminology used in guide tables of content was generally NetWare and Novell specific.

An alternative was to search novell.com. However, the Novell site's search capabilities are generally acknowledged to deliver, at best, only marginally useful results.

Fortunately, Novell customers had been using NetWare for at least a decade and were well acquainted with its services and their management tools. They knew Novell terminology and were familiar with the documentation for the components they used. And if they got really stuck, they would log a service request or download the PDF version of a few guides and use the Acrobat search functionality until they found what they were looking for.

A New Challenge

Open Enterprise Server (OES) brought NetWare services to the Linux platform, and Novell set its sites on growing the customer base.

As the OES information architect, Larry Tomkinson had first-hand experience with finding NetWare service information as a "new user."

After being away from NetWare details for more than a decade, he was assigned in 2003 to develop the documentation for Novell Nterprise Linux Services (NNLS), Novell's first product to feature NetWare services running on Linux.

The NNLS experience taught Larry that finding NetWare service information as a new user was a formidable task. After struggling to come up to speed for NNLS, he had a renewed conviction that information access was critical to OES success, and that, although certainly useful, providing a list of guides or even groupings of guide titles would not be adequate for users of this essentially brand-new product.

Working with Rich Ballard, team members, Novell Support, and a couple of System Engineers, Larry expanded on the service topics that had already been identified in the NetWare online TOC. Then he developed what he hoped would be a logical way to present task information—a multi-level hierarchy designed to lead users to the tasks they needed.

OES 1 TOC: Limited Success

The multi-level hierarchy approach, although fine in theory, didn't work well in practice.

Users responded positively to the *Getting Started* page and a small *A-Z Task List*, but the hierarchical approach to locating task links wasn't usable.

A Paradigm Shift

When OES 2 planning started, Denise Jewkes found an article by Jared Spool, a well known usability expert. A couple of statements in the Spool article (shown below) were the catalyst that led to Larry's rethinking the overall TOC design.

> Nobody starts their design with the objective, "We need our home page to be as complex as we can possibly make it." On the contrary, everybody wants to build simple designs. Yet, somewhere along the line, simplicity translated into "Provide as few links on the home page as possible."
>
> News sites understand [the importance of being "link rich"]. One could imagine a home page for a news site with only these links:
>
> - Top News
> - Local News
> - Sports News
> - Financial News
> - Funny News
> - Brad Pitt / Angelina Jolie News
>
> On a simplified news site, that's all that would be required. It would make home page maintenance a dream. You'd never have to change it.
>
> However, it wouldn't make for a good news reading experience because users want to see what headlines are under each category. Instead [of presenting only a few links, news sites] make the news transparent by providing a link-rich environment.
>
> *Lifestyles of the Link-Rich Home Pages*
> June 15, 2006
> © Jared Spool

OES 2 documentation users needed a direct path to information. They needed link-rich pages that could get them to target information within a couple of mouse clicks. They needed pages that were, in some ways, similar to OES 1's popular *Getting Started* and *A-Z Guides List* pages.

Larry immediately undertook a reworking of the OES 1 TOC pages, and with Rich's help after a couple of weeks had a redesigned OES 1 TOC rolled out.

B.1.2 The Current OES 2 TOC

- "A Fresh Start" on page 57
- "Standard OES 2 TOC Page Breakdown" on page 58
- "Variation 1—No Subtopics" on page 60
- "Variation 2—A to Z Lists" on page 61

A Fresh Start

- "Getting Everyone on the Same Page" on page 57
- "Approval and Rollout" on page 58

Getting Everyone on the Same Page

After the OES 1 rework was online, the OES 2 doc team held a number of planning meetings to

- Examine and critique Larry's redesigned OES 1 TOC.
- Discuss the categories (topics) that Larry had identified, adding a couple in the process.

- Explore and group all aspects of OES 2 components and services under the topics.
- Assign each book to a primary category.
- Work out a book filenaming convention that indicates the category, component/service name, and platform covered by each book.

  This was something that Catherine Craft had suggested for the OES 1 releases, and it has proven to be extremely valuable whenever quickly identifying a book from the filename is required, for example, during link creation.

Approval and Rollout

After some layout modifications that the documentation managers requested for consistency with other Novell products, the final OES 2 TOC design was approved. It complied with the Novell documentation TOC layout standard at that time and was also unique in some respects among Novell product online TOCs, most of which consisted of a single HTML page.

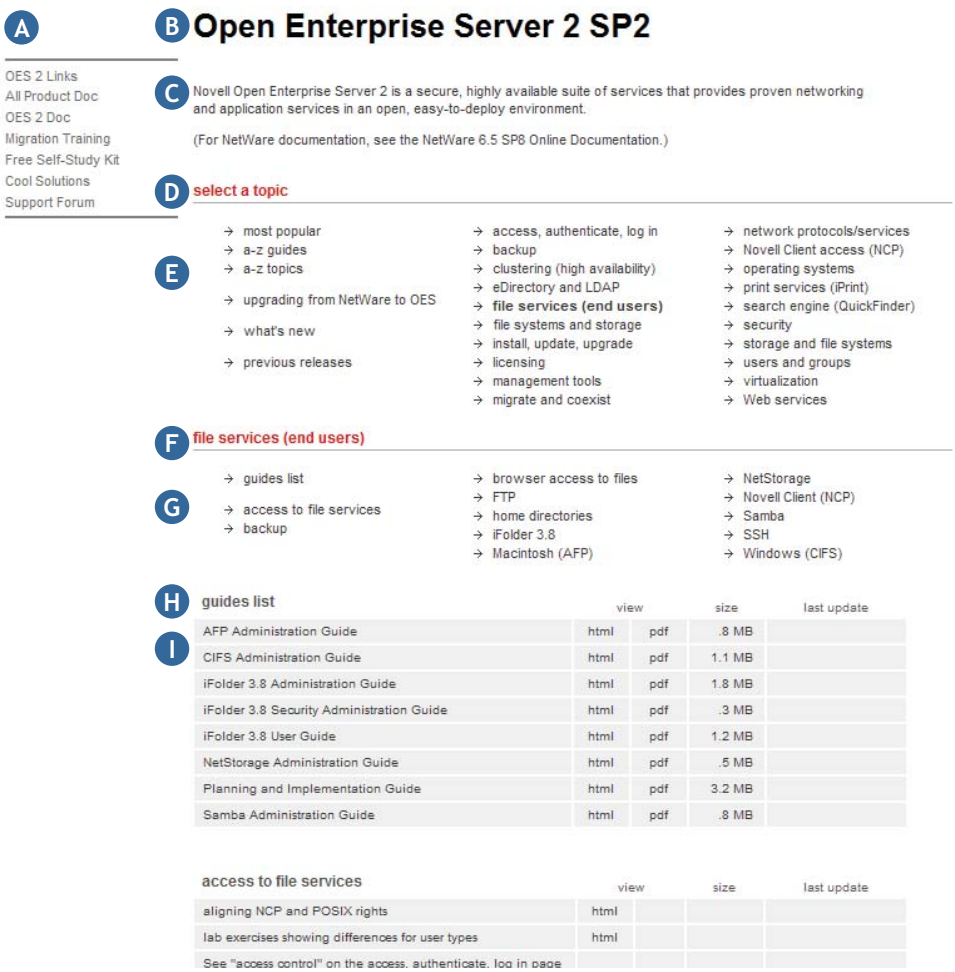
Since the initial OES 2 release, users have suggested refinements, such as providing a list of guides on each topic page, but there have been no complaints or suggestions regarding the overall design, and it has proven to be very flexible as product needs change.

OES 2 TOC source files consist of Structured FrameMaker files in a FrameMaker 7 book that is part of the OES 2 documentation set. Because of the DocSys toolset, the maintenance burden is quite lite. Without the DocSys toolset, it would be impossible to maintain so many links and HTML pages.

NOTE: After OES 2 TOC approval, the OES 1 TOC was updated with the approved OES 2 look and feel.

Standard OES 2 TOC Page Breakdown

Each OES 2 TOC page is organized into different sections as indicated by the letter notations on the following screen shot of the File Services page.



Section	Name	Description
A	Left Sidebar	Links related to OES, including a link to a Documentation Search page.
B	Product Name	The OES product name.
C	Product Description	An abbreviated version of the Novell Marketing's product description.
D	*select a topic* Heading	This title is the same on every page.

Section	Name	Description
E	Links to Topics	A set of navigational links that is the same on every page, except that the currently selected topic (or page) is bolded. The left column contains links to special-purpose, product-level pages, such as Most Popular, A-Z Guides, A-Z Topics, and Previous Releases. It also contains direct links to select project guides and topics. For OES 2 these are currently the *OES 2 SP3: Upgrading to OES—Best Practices Guide*, and the "What's New or Changed" section in the *OES 2 SP3: Planning and Implementation Guide*. The middle and right columns contain alphabetically listed links to the defined OES service topics or "buckets" as they're sometimes called, such as File Services, Management Tools, and Security.
F	*topic* Heading	This shows the currently selected topic and is the highest element on the page that changes as different topics are selected.
G	Links to Subtopics	This set of navigational links exists when the page includes subtopics. It ◆ Indicates the various subtopics that, on most pages, are not all visible at the same time due to their length. ◆ Provides quick access to those subtopics.
H	*subtopic* Headings	The topic heading on most pages is followed by a series of subtopics that "chunk" and organize the information within the topic. For example, the File Services page includes links to sections that ◆ Cover the various file services in OES (FTP, iFolder, Macintosh (AFP), NetStorage, Novell Client (NCP), Samba, SSH, and Windows (CIFS)). ◆ Present subtopics that customers have said they expect to see on the File Services page, such as the Guides List, Access to File Services, Backup, Browser Access to Files, and Home Directory Information.
I	Target-Related Links	These sets of links are the heart of the TOC. They link directly to the sought-after information and/or instructions. And unlike many of the section titles within the guides themselves, they use industry-standard and user-requested terminology.

Variation 1—No Subtopics

Some topics don't require subtopics.

Before we received the user request for a list of relevant guides on each topic page, there was a handful of pages that didn't require the second topic layer. For SP3, there are only two:

- most popular (http://www.novell.com/documentation/oes2/index.html#bktitle)
- previous releases (http://www.novell.com/documentation/oes2/previous-releases.html#previous-releases)

Variation 2—A to Z Lists

When users want direct access to particular guides, the a-z guides page (http://www.novell.com/documentation/oes2/allguides.html#allguides) is the quickest route.

Some users prefer an alphabetical listing of the most common tasks and topics as presented by the a-z topics list (http://www.novell.com/documentation/oes2/athruz.html#athruz).

The a-z list pages are patterned after the Novell Documentation A-Z page, with the important variation that rather than grouping the access letters (A-C, D-F, etc.), each letter is a separate link. This is because initial attempts to group letters together resulted in lists whose length made it difficult to find anything in them.

B.1.3 About the TOC Template

- "Template and Tool Modifications" on page 61
- "Style Notes" on page 61

Template and Tool Modifications

The following changes were made to the Frame 7 template and toolset for the reasons listed

Table B-1 *Frame 7 Doc Template and Tool Changes Required for the TOC Template*

Change	Reason
Page length is 16.5 inches.	This was required by an earlier version of the template wherein table cells were too tall to fit on one page and would scroll off. The reason for the non-standard page length no longer exists, so retaining it should not be necessary if the template is ever replaced by an official Novell Frame TOC template.
EDD changes	Two reasons: ◆ Accommodate elements in places that they are not normally found. ◆ Facilitate the UltraEdit macro-based conversion of the standard HTML output to the documentation TOC standard.
DTD changes	Compatibility with the EDD.
HTML generation change	Each chapter and its content produces a single HTML page. First-level Head elements do not produce separate files.

Style Notes

- All entries are lower-case unless capitalization is required. In other words, the title caps rules don't apply in the OES TOC.

NOTE: While preparing this section, Larry noticed that some entries in the current OES 2 TOC don't follow these conventions. These have been corrected in the Frame source files and will be republished before the Public Beta release of SP3.

B.1.4 About the TOC Macros

The TOC macros are critical to TOC production and are currently maintained by the OES information architect, Larry Tomkinson.

Because of a macro corruption issue that occured in early versions of UltraEdit 32, each macro now has an associated text file. That way if a macro becomes corrupted, it can be easily replaced. Always modify the text file and then replace the macro text using copy/paste from the text file. Never modify a macro statement by editing the macro directly.

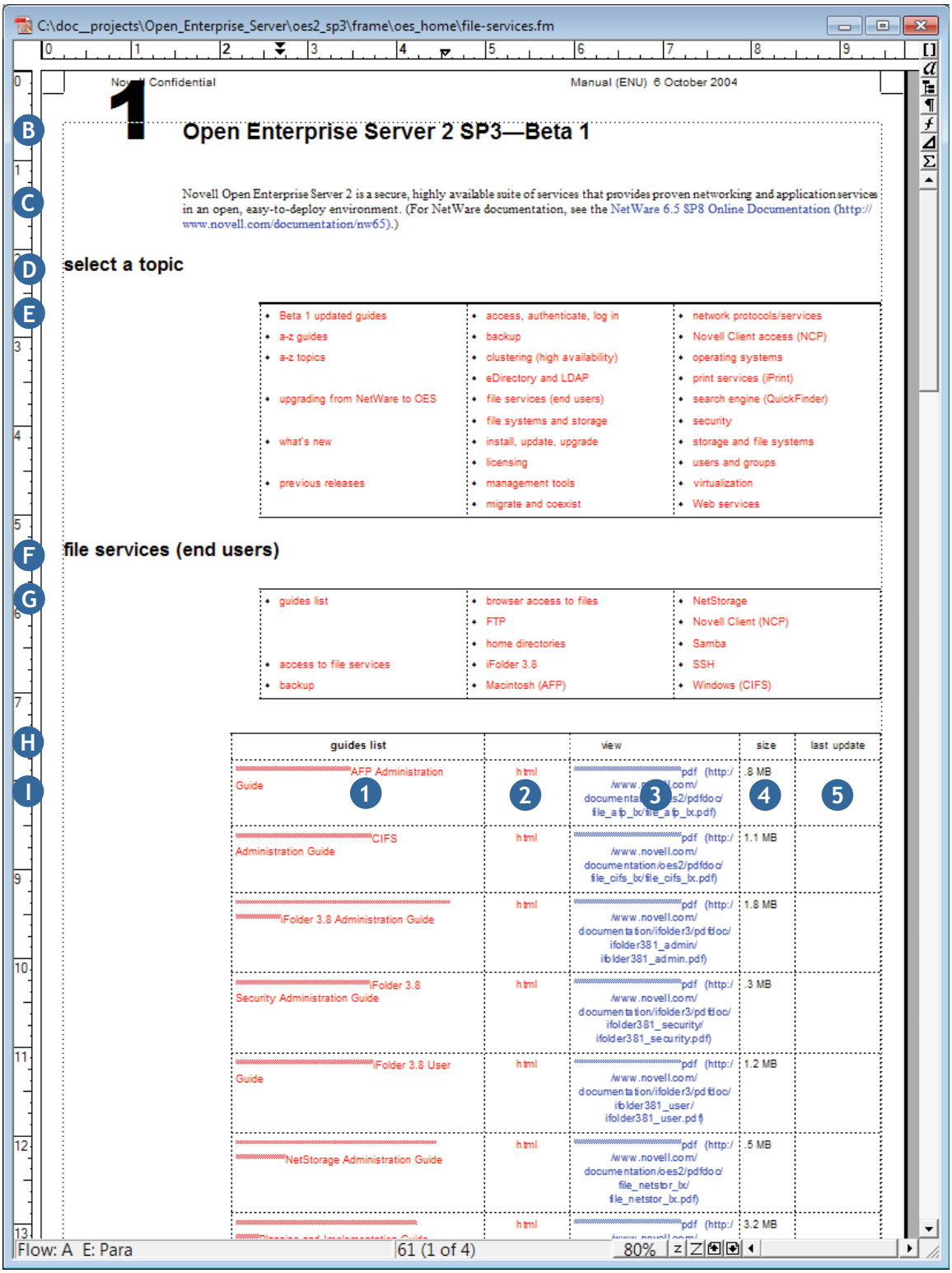
NOTE: The hope is that at some point there will be a standard NovDoc template and production tool that can replace the Frame 7 template and UE macros while not compromising the flexibility and power of the OES TOC.

B.2 Working in the TOC Book

- Section B.2.1, "Authoring a Standard TOC Topic (Chapter)," on page 62
- Section B.2.2, "Authoring Chapters with No Subtopics," on page 66
- Section B.2.3, "Authoring an A–Z Chapter," on page 67
- Section B.2.4, "Focus on Helping OES Administrators," on page 67
- Section B.2.5, "Linking Guidelines," on page 68
- Section B.2.6, "Alphabetizing the TOC Tables," on page 70
- Section B.2.7, "Preserving Link Target Integrity Is Critical," on page 70

B.2.1 Authoring a Standard TOC Topic (Chapter)

The Frame TOC template follows the same organization as the HTML page it ultimately produces.


A
C:\doc_projects\Open_Enterprise_Server\oes2_sp3\frame\oes_home\file-services.fm
Novell Confidential
Manual (ENU) 6 October 2004
B
Open Enterprise Server 2 SP3—Beta 1
C
Novell Open Enterprise Server 2 is a secure, highly available suite of services that provides proven networking and application services in an open, easy-to-deploy environment. (For NetWare documentation, see the NetWare 6.5 SP8 Online Documentation (http://www.novell.com/documentation/nw65).)
D
select a topic
E
Beta 1 updated guides
a-z guides
a-z topics
upgrading from NetWare to OES
what's new
previous releases
access, authenticate, log in
backup
clustering (high availability)
eDirectory and LDAP
file services (end users)
file systems and storage
install, update, upgrade
licensing
management tools
migrate and coexist
network protocols/services
Novell Client access (NCP)
operating systems
print services (iPrint)
search engine (QuickFinder)
security
storage and file systems
users and groups
virtualization
Web services
F
file services (end users)
G
guides list
access to file services
backup
browser access to files
FTP
home directories
iFolder 3.8
Macintosh (AFP)
NetStorage
Novell Client (NCP)
Samba
SSH
Windows (CIFS)
H
I
guides list
view
size
last update
AFP Administration Guide
html
pdf (http://www.novell.com/documentation/oes2/pdfdoc/file_afp_bx/file_afp_bx.pdf)
.8 MB
CIFS Administration Guide
html
pdf (http://www.novell.com/documentation/oes2/pdfdoc/file_cifs_bx/file_cifs_bx.pdf)
1.1 MB
iFolder 3.8 Administration Guide
html
pdf (http://www.novell.com/documentation/ifolder3/pdfdoc/ifolder381_admin/ifolder381_admin.pdf)
1.8 MB
iFolder 3.8 Security Administration Guide
html
pdf (http://www.novell.com/documentation/ifolder3/pdfdoc/ifolder381_security/ifolder381_security.pdf)
.3 MB
iFolder 3.8 User Guide
html
pdf (http://www.novell.com/documentation/ifolder3/pdfdoc/ifolder381_user/ifolder381_user.pdf)
1.2 MB
NetStorage Administration Guide
html
pdf (http://www.novell.com/documentation/oes2/pdfdoc/file_netstor_bx/file_netstor_bx.pdf)
.5 MB
html
pdf (http:/
3.2 MB
1
2
3
4
5
Flow: A E: Para
61 (1 of 4)
80%

Section	Name	Description
A	Left Sidebar	The left sidebar is not authored in the Frame Template. The reference to it is inserted in each HTML page by the first UltraEdit macro that runs.
B	Product Name	This is contained in each Chapter Title and doesn't visually change as each HTML page loads. The chapter number is not retained in the HTML. This title, the product description, and the first Head Title (*select a topic*) are the same in each chapter so that the top part of the page appears to be static.
C	Product Description	This is a Para element. The text must be the same in every chapter so that the top part of the HTML page is visually static from page to page. Since the release of SP2 and the general removal of NetWare from OES, it contains a ULink to the NetWare 6.5 SP8 Online Documentation.
D	*select a topic* Section	This is a Head element and would be a Sect 1 in the current template. It is the same in every chapter.
E	Links to Topics	This table has three cells with a bullet list in each. The links in each bullet are XRefIntText links. This is required for the UltraEdit macros to function correctly. Bullet Item insertion and alphabetic ordering is a manual process. Although the text in most of the targets is exactly what appears in the table, having a "Text" link provides important flexibility for customizing the end user experience. For example, the *file systems and storage* and *storage and file systems* links actually link to the same chapter, but both entries are required because some users look for "storage" and some look for "file systems." Two of the links (*upgrading from NetWare* and *what's new*) link directly to guides rather than TOC chapters because there was no need to require an interim step for reaching the target.
F	*topic* Section	This is a structural peer to *select a topic* and could be thought of as the actual "title" of the Chapter because everything that preceeds it is the same throughout the book.

Section	Name	Description
G	Links to Subtopics	The topic heading on most pages is followed by a series of subtopics that break down and organize the information within the topic. This table provides links to those subtopics and consists of three cells with a bullet list in each cell. The links in this table are XRefInt links. This is required for the UltraEdit macros to function correctly and ensures that changes to the target text are reflected in the table. Bullet Item insertion and alphabetic ordering is a manual process. Each entry links to a ParaWithTitle element found in the upper left cell of each subtopic table. The UltraEdit macros turn these ParaWithTitle elements into the subtopic headings. The entries in the table must be manually arranged so that the table reflects the order of the subtopics on the page and is visually balanced. ◆ The upper-left bullet link always points to the *guides list* table that is the first table on each standard page. ◆ The *guides list* entry must be followed by at least one blank vertical space (two linebreaks) to separate it from the alphabetical listing of subtopics that follows. ◆ For pages with 8 or fewer subtopics, the alphabetically ordered subtopic bullet items appear in only the middle and right columns. ◆ If there is an uneven number of entries, the extra entry should be placed in the middle column. For example, see the Novell Client access (NCP) (http://www.novell.com/documentation/beta/oes2/novell-client-access.html#novell-client-access) page. ◆ For pages with more than 8 subtopics, the bullet list starts in the the first column, two linebreaks below the *guides list* entry. For example, see the file services (end users) (http://www.novell.com/documentation/beta/oes2/file-services.html#file-services) page.
H	*subtopic* Sections	Each subtopic section contains a series of alphabetically ordered links to tasks and other important information for the subtopic. These sections are where “the rubber meets the road” as it were. They leverage common gerund entries, such as understanding, installing, maintaining, and troubleshooting, as well as industry-standard terms to help users find links to what they are seeking as quickly as possible. Each table row contains five columns and potentially five entries.

Section	Name	Description
I	Target-Related Links	The five columns in this table correspond to the columns in the online view. As with the rest of the template, it is important to follow certain conventions when entering the information in each cell so that the UltraEdit macros function correctly. The following guidelines apply: 1. This cell contains a Para element that in turn contains an optional Superscript element followed by either an XRefExtText element for links into Frame documents or a ULink element for non-Frame links, such as information on the Web. The Superscript element is a holdover from the first OES 1 TOC and appears only with links to guides. It contains a brief description of guide content that shows up as mouse-over text in the online TOC. The reason this was implemented was that it isn't always possible to tell what a guide contains or what its purpose is from its title. This was especially true for the NetWare documentation set. Alternatively, this Para element can contain a text explanation or instruction, such as just-in-time information. See Section B.4.3, "Providing Just-in-Time Information," on page 73. 2. This cell contains a duplicate link from the first cell, but the link text is replaced with "html". The resulting link and its placement in the table comply with the doc standard design. 3. This cell is only used when the row target is an entire Frame guide, and it contains a ULink to the PDF version of the guide. The Superscript is another holdover from the OES 1 days and states, superfluously, "Download a PDF version of the guide." The resulting link and its placement in the table comply with the doc standard design. 4. This cell contains text indicating the approximate size in MB of the PDF file. 5. This cell contains the date when a guide was last published or updated. <fix the TOC so this is correct.)

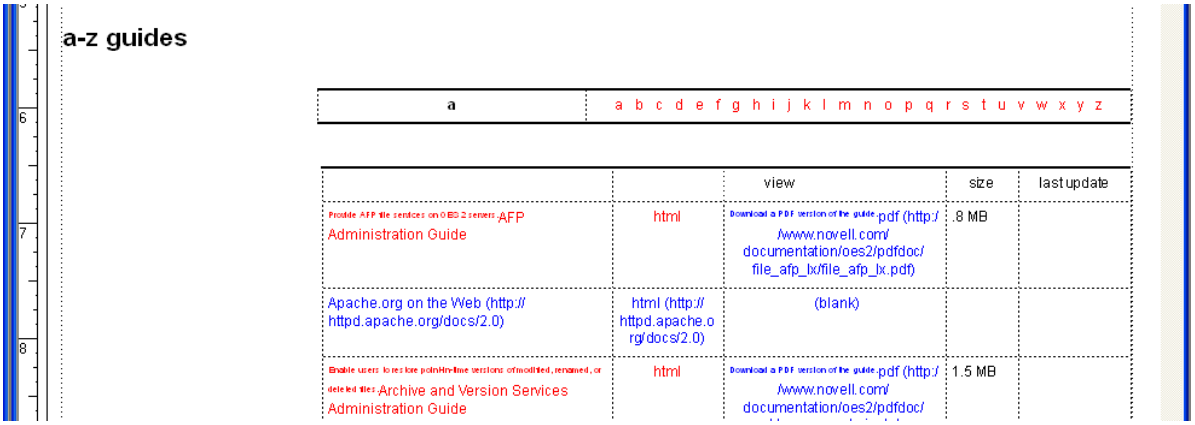
B.2.2 Authoring Chapters with No Subtopics

For OES 2 SP3, only two topic chapters don't require subtopics: the most popular page (http://www.novell.com/documentation/oes2/index.html#bktitle) and the list of previous releases (http://www.novell.com/documentation/oes2/previous-releases.html#previous-releases).

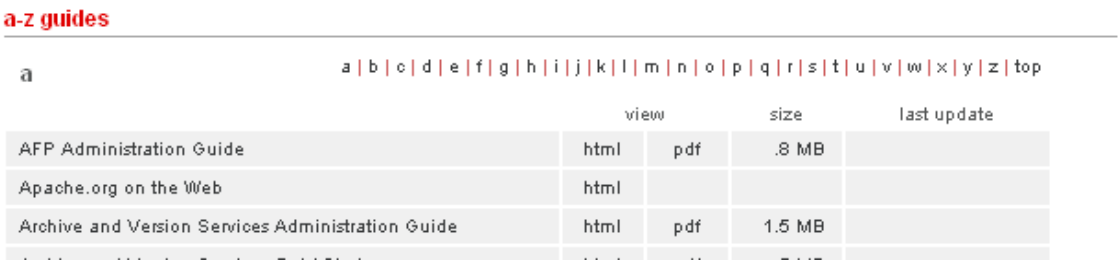
Both sections contain flat lists of links that are all peers, subject wise, and don't need further categorization for accessibility purposes.

B.2.3 Authoring an A–Z Chapter

The *a-z guides* and *a-z topics* chapters are the same as the standard topic chapters, except that the alphabet-letter subtopic headings are contained in a separate one-row table. The first cell contains the letter for the section. The second cell contains XRefInt links to each of the first-cell letters. Each XRefInt link is separated from the next by three empty spaces.



The UltraEdit macros convert this series of links to the letter links that separate each section of the online page as shown below.



B.2.4 Focus on Helping OES Administrators

When working with the TOC, do the following:

1. After submitting the initial revision or creation of a section to Larry, leave it alone for one or two days and work on something unrelated while Larry implements your input.

   Then, when you have sufficient time to do the remaining steps in one sitting, continue with Step 2.

2. Imagine that you are an administrator who needs information or instructions on the topic you just worked on.

3. Think of a situation the administrator might be in and the information he or she might need to find.

   Then, using administrator jargon and industry-standard terminology as much as possible, write down what the administrator needs to find. For example, “I need to know what the service differences are between NetWare and OES Linux.”

   Repeat until you have up to six items, or until idea generation lags, whichever happens first. Then continue with Step 4.

4. Try to locate at least one link for each item on your list.
   - 4a Note each failed attempt so that you can evaluate whether to provide additional links.
   - 4b If you can't find the information in three tries or less, move to the next item. Your notes on three failed attempts should be sufficient to remedy the situation.
5. After working through all of the items, specify the details for links that would have led you to the information you needed to find, then ask the OES information architect (currently Larry) to add or revise the links according to your requests.

NOTE: Larry has requested that the TOC pages be instrumented for User Comments. That would make requesting new and modified links must quicker and simpler. Until that happens, however, a list of changes in e-mail is the preferred option.

This experience will

- Become easier each time you go through it.
- Help to improve your writing.
- Help your users to quickly find what they need.

B.2.5 Linking Guidelines

Requests for adding links must comply with the following guidelines.

- Link to the logical parents of important topics.

  These are usually chapters with titles such as "Configuring *XYZ*". Avoid linking to individual tasks within parent topics unless you know that the task is a trouble spot for many administrators, that they will look for it directly, and that they won't think of it in association with the parent topic.

  As a service owner, it is easy to believe that many topics/tasks need special emphasis, so it is critical to keep a balanced and detached perspective. Otherwise, the TOC could quickly expand to the point that it becomes unusable.

  NOTE: A customer request for a link to a specific topic/task warrants including the link requested.

- Use as few words in links as possible.
- If applicable, use gerunds to summarize the link target contents, such as
  - Administering
  - Configuring
  - Implementing
  - Installing
  - Maintaining
  - Planning
  - Securing
  - Setting Up
  - Troubleshooting
  - Understanding

- Choose the most precise word. For example, “Setting Up” implies setting up multiple components to create an environment, whereas “Installing” implies only an installation procedure.
- When linking to multiple topics that have the same gerund in the title, include a key differentiating term.
  - Installing XYZ on OES
  - Installing XYZ on Windows
  - Managing iFolder Users
  - Managing iFolder Servers
- In some cases, separate guides indicate that separate TOC subtopics are needed, but this is not always the case. Weigh the importance of the components against each other from an administrative perspective.

  For example. iPrint includes a Health Monitoring feature that has its own guide. But iPrint as a general topic and iPrint Health Monitoring are certainly not peers to each other. Therefore, the iPrint section contains a single link to the Health Monitoring guide, and the result is that Health Monitoring (a guide) is appropriately listed as a peer to Installing (a chapter).
- In some cases, components are of equal weight but don't warrant separate subtopics

  For example, the iFolder 3 Enterprise Server and iFolder 3 Web Access Server are basically peers in the iFolder 3 subtopic.

  A single “Installation” link covers both server types, but there are separate “managing” chapters for various, equally important parts of the service. Therefore, there are links in the TOC to each management chapter.
- Most products have unique aspects that are documented outside of the standard Installing, Planning, and Maintaining buckets. Linking to these chapters requires that one step back and think about what best communicates the content of the chapter.

  For example, QuickFinder includes a feature named “Virtual Server,” but unless you are familiar with QuickFinder, you can only guess at what a virtual server is. Therefore, the TOC has a link named “Searching multiple indexes,” which is what having virtual QuickFinder servers is actually all about.
- Add, create, and/or modify links in the TOC only after you are certain the guide organization is stable.
- Print a copy of each guide's TOC, then click through the OES TOC to see which sections have links and which need links to them.
- Use only XRefExtText Frame links when possible. These take advantage of Frame linking and also let you customize link words.

  Use ULinks only for information that is not contained in a Frame collection, such as a Web site.
- Do not copy table rows to create new link rows. It's too easy to forget to update the HTML and PDF links. Instead, create a new blank row for each link. The blank spaces will remind you to also create HTML and PDF links as required.

  For blank cells, be sure to match the content of the other blank cells in the same column. The UE macros depend on certain element tags always being present in each cell of a column, even if the cell is blank. An easy way to ensure correct cell content is to copy a blank cell in the corresponding position from an existing row to the new row.
- On most of the TOC pages (chapters), you should create only one link to a target.

However, the *a-z topics* page is an important exception. Administrators look on this page to find topics as they would personally describe them.

For example, three administrators might think of three different terms in connection with Microsoft workgroup file services. One might look for Samba, another for Windows, and yet another for CIFS. Therefore, all three terms should be included in the *a-z topics* page.

B.2.6 Alphabetizing the TOC Tables

After creating or adding to the links in any section *except the a-z guides chapter*, do the following:

1. Select all the link rows (not the header row) in a given table.
2. Click *Table > Sort*.
3. Sort the list on column 1.

IMPORTANT: The *a-z guides* list must be alphabetized manually.

This is because the first column has a superscript element first that the UE macros convert to mouse-over text. Using the sort feature in this chapter would alphabetize the mouse-over text entries rather than the link text.

B.2.7 Preserving Link Target Integrity Is Critical

The most time-consuming task for maintaining the TOC involves resolving broken links caused by filename and ID changes. For this reason, all team members must avoid the following when possible.

- Changing product, book, or file names.
- Changing section IDs.
- Creating new chapters or sections as replacements for sections linked in the TOC.

If doing one of the above cannot be avoided, you must inform the OES information architect (currently Larry Tomkinson) of the changes.

B.3 Producing the TOC

IMPORTANT: Although there is a Production Group effort to create a TOC template, TOC maintainers must currently use the OES-customized October 2004 DocSys7 toolset to work with the OES TOC book (oes_home).

The macros referred to in this section and their associated text files are located in the doc SVN repository for each OES project under *tools (a peer to the frame directory) > TOC > project TOC Macros*.

Be sure to use the macros associated with a specific project because small tweaks are usually required for different release versions.

To produce a new build, do the following:

1. Save the book as SGML.
2. Copy the SGML file to a build directory, such as `c:\builds\english\oes`.
3. Open a DOS shell.

4 Change to the NovDoc directory for the October 2004 DocSys7 toolset. For example,

```
cd c:\DocSys7\NovDoc
```

5 Run the build process by entering the following command:

```
buildwebhtml target\build\directory\path
```

For example,

```
buildwebhtml c:\builds\english\oes
```

This creates an oes_home\data directory that contains all the HTML files.

6 Open UltraEdit 32.

7 Delete both the `toc*` files.

8 Open all the files in the `oes_home\data` directory, except `legal.html`.

9 Load the `01_replace_all_wrappers` macro.

10 Run the macro on each open file by pressing Ctrl+m. This adds style sheet references and other code to the front and back matter, saves the file, and closes it.

11 Load and then run each of the following macros in turn using Ctrl+m:

- `02_remove_table_titlesa.mac`
- `03_remove_table_titlesb.mac`

12 Perform a Search and Replace on all .html files in the `\data` directory to put the table rows and cells on the same line, as follows

12a Search for `>^p<TR` and replace with `><tr`

12b Search for `>^p<TD` and replace with `><td`

13 Load and run (Ctrl+m) each of the the following macros in turn:

- `05b_clean_up.mac`
- `05c_format_rows.mac`
- `06_a-z_front.mac`
- `07_a-z_allguides.mac`
- `08_top_table_align.mac`
- `09_adjust_remaining_rows.mac`
- `10_emphasize_selected.mac`
- `11_fix_pdf_links.mac`

14 Delete the *.bak files in the /data directory (if UE is configured to create them).

15 Commit the changed files to Subversion.

16 After the autobuild happens, verify the update on the novell.com testing server (http://wwwtest.provo.novell.com/documentation/oes2), then log a production bug to have the TOC updated on the appropriate external site.

B.4 The OES TOC Is Also an Important Tool

The OES 2 writing team and especially the information architect can and should use the TOC as a tool to improve the OES 2 documentation set.

- Section B.4.1, "Aiding the Migration from NetWare to OES," on page 72
- Section B.4.2, "Monitoring for Consistent and Complete Coverage," on page 72
- Section B.4.3, "Providing Just-in-Time Information," on page 73

B.4.1 Aiding the Migration from NetWare to OES

The OES 2 TOC is an important tool for providing information parity for our NetWare customers, both internally and externally, especially in those cases where gaps are inherent to the OES development decisions that Novell has made, such as:

- **Expected Services:** We know that some NetWare services will never be available on OES Linux.

  For example, Novell Licensing Services (NLS) doesn't fit with the GNU licensing restrictions that apply to OES Linux. NetWare administrators who look for a *Novell Licensing Services Administration Guide* won't find one in OES. Nevertheless, they can quickly find links in the TOC to sections that explain how OES licensing works.
- **Closed vs. Open Source:** Some services, such as Apache and OpenSSH had Novell guides for NetWare, but for Linux they are documented on the Web. The documentation team is charged with identifying and publishing where people can get information about these services. The TOC provides the primary location for these links.
- **Information Gaps:** The TOC helps the documentation team, especially the information architect, check for information gaps by asssessing whether all of the expected topics (planning, installing, etc.) are covered (see Section B.4.2, "Monitoring for Consistent and Complete Coverage," on page 72) and also by comparing the topic links for different services against each other.

  When a possible gap or oversight is found, we can talk with developers and product managers about providing information parity. We then have three basic options:

  - Enhance the existing documentation.
  - Link to resources on the Web.
  - Acknowledge the gap and explain why it exists.

B.4.2 Monitoring for Consistent and Complete Coverage

OES 2 administrators should be able to find information on at least the following topics for each OES 2 service and component:

- Understanding
- Planning
- Installing
- Maintaining

The information architect uses the TOC to verify that these topics are linked and available for each service.

If they are not available, the architect contacts the responsible writer for the location of the needed information. If one of the topics, for example planning, doesn’t apply, then that should be noted in the TOC so that OES administrators don’t spend time searching for information that doesn’t exist.

B.4.3 Providing Just-in-Time Information

Some topics, such as file lockouts and USB installation procedures are not covered in the OES documentation.

Administrators shouldn't need to look beyond the TOC to realize that they can find information about lockouts in the iManager help, and for USB device instructions they need to use the documentation that came with the device.

Elevating this kind of information to the online TOC helps to prevent users from wasting time searching for information that doesn't exist in the online documentation.

B.5 Looking Forward to Ponderosa

It is vital that we establish good practices for OES 2 SP3 because that will lay the foundation for Ponderosa, which is the next release of OES and a critical product for the future of Novell.

The Writing Team can help prepare for Ponderosa by becoming familiar with the information in this chapter and becoming actively involved in helping to maintain the TOC.

User Comments feedback is also critical, and so Larry has logged Bugzilla Defect # 629225 to request that User Comment support be added for the OES TOC pages.

C Maintaining the OES Readme

Because the OES readme is so large and complex, it doesn't use the standard Frame readme template.

The readme evolves over the course of the project. Some sections are needed for only one or two Beta releases. Some are needed for all Beta releases and then transitioned to the regular documentation for FCS. Some are part of every product release.

After Beta 1 is released, two versions of the readme must be tracked throughout the rest of the project lifecycle until the product is released.

Before Beta 1, do the following:

1. Prepare the documentation work area for the project and include the `oes_readme` book directory in the `frame` directory.
2. Open the book and all of the chapters in Frame.
3. Add all of the chapters into the book that were not included in the FCS readme.
4. As appropriate, change the version number in the front and other files.
5. Prepare the What's New chapter for Beta 1 by removing everything except an appropriate introductory paragraph and the Sect1 Element and its associated Title element for the "New in Beta 1" and "New in eDirectory xxx" sections.
6. Prepare the "Caveats" chapter by replacing the content of each section with an empty Para element.
7. Prepare the "Before You Install or Upgrade" chapter.
   - **7a** Make a copy of it with the Upgrade sections included and indicate its purpose in the directory name.

     You will need this information when it comes time to develop the readme for the Beta release that supports upgrading.
   - **7b** Commit the chapter to Subversion.
   - **7c** Update the non-upgrade sections based on what you know about the product thus far.
   - **7d** Remove the Upgrade sections.
8. Prepare the "Service Issues," "Migration Tool Issues," and "Domain Services for Windows Issues" chapters by checking each Bug number in Bugzilla and removing the items that have been fixed.
9. Prepare the "Documentation Updates" chapter.

NOTE: In contrast with Documentation Updates appendixes in other guides that are cummulative, this is included in the guide only when there have been changes to the last Beta or FCS release. Tracking what changed in the readme for each previous release clutters the section and doesn't help customers.

9a Remove all Bridgehead sections but one.

9b Change all text, except the introductory paragraph, to a placeholder to remind you of what goes where when a change is made.

D Profiling in OES 2 SP3

Some Open Enterprise Server 2 SP3 (OES 2 SP3) documentation team members wanted to start using release-profiling for OES releases, starting with the SP3 release. However, after some discussion, the team decided to postpone making a decision until the potential benefits and problems associated with the change are thoroughly discussed and understood.

Assuming that such a change might be suggested in the futurre, this section outlines the requirements and strategy for implementing this change.

- Section D.1, "Historical Perspective," on page 77
- Section D.2, "Why the Proposed Change?," on page 77
- Section D.3, "Open Enterprise Server Is a Unique Novell Product," on page 77
- Section D.4, "Challenges to Overcome," on page 78
- Section D.5, "Plan for Overcoming," on page 79

D.1 Historical Perspective

To this point in time, each OES product release has a version-specific work area or repository on a commonly accessible server. All of the files associated with that product release version are stored in the common area, including files that are created or modified after the initial product release.

The implementation of profiling within the NovDoc tool set has made it possible to store the source files for multiple versions in the same repository.

D.2 Why the Proposed Change?

Some doc team writers want to maintain only one set of source files for OES 2 SP2 and OES 2 SP3.

For example, if information changes regarding SP2, and that change also applies to SP3, writers want to make the change only in one set of source files. Under the version-specific repository model, changes made in the SP2 set would need to be copied to the corresponding location in the SP3 set.

D.3 Open Enterprise Server Is a Unique Novell Product

- Section D.3.1, "Size," on page 78
- Section D.3.2, "Service Diversity and Complexity," on page 78
- Section D.3.3, "Link Density," on page 78
- Section D.3.4, "Flagship Status," on page 78

D.3.1 Size

OES is larger than other Novell products. Even with the removal of NetWare from the product, the documentation set still includes over 30 OES-specific guides, and when the bundled products, such as iFolder 3, iManager, and eDirectory are included, set volume is larger than almost all other Novell products by an order of magnitude.

D.3.2 Service Diversity and Complexity

OES is also more diverse and hence more complex than other Novell products. This is illustrated by the fact that the OES 2 documentation site identifies about 20 different service areas that administrators must manage.

D.3.3 Link Density

OES administrators need to be able to find needed information quickly. To facilitate this, OES includes an online TOC that is, again, larger than any other product's TOC by at least an order of magnitude. The set also includes a planning and implementation guide with links into the OES and bundled product guides. The TOC contains almost 4,500 links; the P&I guide about 1,500.

D.3.4 Flagship Status

As the successor to NetWare, OES is Novell's flagship product. Many enterprises have depended on NetWare and must now depend on OES to run their businesses. Consequently, product visibility and possible legal liability issues are greater.

D.4 Challenges to Overcome

NOTE: The following explanations were authored early in the SP3 product life cycle and although outdated are still relevant because they illustrate the challenges for future releases if profiling is considered again.

Achieving the benefit (see Section D.2, "Why the Proposed Change?," on page 77) that release-profiling can bring to the OES 2 documentation effort will require overcoming the following challenges:

- **SVN Repository Confusion:** The repository for the current release of OES 2 is the `oes2_sp2` folder in SVN. For the next OES 2 release, it might be renamed to `oes2_sp3`, or the contents might be copied to a new `oes2_sp3` folder or an `oes2_spx` folder, etc.

  For legal protection, there must be a fail-safe method for indicating in the SVN repository itself, the respective date boundaries so that an SP2 documentation set that is specific to any given point in time can be generated with certifiable accuracy from the correct repository. There must be some way to indicate to someone in a future time how and from where one must generate the SP2 set for a given date in the past.
- **Profiling Guidelines for All Guides:** Writers must have clear instructions for preparing each of their guides to function in the release-profiled environment. Three guide types come to mind:
  - Guides with both SP2 and SP3-profiled content.

    - Guides with only SP3-profiled content (no SP2-specific instructions need to be separated out).
    - Guides that have a companion SP2 version in the repository.
- **Regular SP2 Validation:** As the SP3 set is developed writers must periodically verify that the SP2 version can still be accurately generated. Changes to content profiled for one release often yield unanticipated results in the other profiles.
- **SP4 Will Compound the Risks:** Developing an SP4 set will involve the same challenges again, but instead of only one previous release being affected, there will now be two.

D.5 Plan for Overcoming

This section will need to be developed and approved by all stakeholders before version profiling can be adopted for an OES release.

E Documentation Updates

This section summarizes the changes made to the documentation plan since its initial publication for OES 2 SP3.

Chapter or Section Changed	Summary of Changes
Throughout	Updated format to comply with new corporate standard.