



Sun Java System Portal Server 7.1 管理ガイド



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820-0867
2007年2月

本書で説明する製品で使用されている技術に関連した知的所有権は、Sun Microsystems, Inc. に帰属します。特に、制限を受けることなく、この知的所有権には、米国特許、および米国をはじめとする他の国々で申請中の特許が含まれています。

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本製品には、サードパーティーが開発した技術が含まれている場合があります。

本製品の一部は Berkeley BSD システムより派生したもので、カリフォルニア大学よりライセンスを受けています。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびにほかの国における登録商標です。

Sun、Sun Microsystems、Sun のロゴマーク、Solaris のロゴマーク、Java Coffee Cup のロゴマーク、docs.sun.com、Java、Solaris は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。Sun のロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。

OPEN LOOK および SunTM Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカルユーザーインターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは、OPEN LOOK GUI を実装するか、または米国 Sun Microsystems 社の書面によるライセンス契約に従う米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

この製品は、米国の輸出規制に関する法規の適用および管理下にあり、また、米国以外の国の輸出および輸入規制に関する法規の制限を受ける場合があります。核、ミサイル、生物化学兵器もしくは原子力船に関連した使用またはかかる使用者への提供は、直接的にも間接的にも、禁止されています。このソフトウェアを、米国の輸出禁止国へ輸出または再輸出すること、および米国輸出制限対象リスト(輸出が禁止されている個人リスト、特別に指定された国籍者リストを含む)に指定された、法人、または団体に輸出または再輸出することは一切禁止されています。

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われないものとします。

目次

はじめに	19
パート I Sun Java System Portal Server の管理	25
1 Portal Server 管理の概要	27
Portal Server コンポーネントの概要	27
Portal Server 管理コンソールの使用	28
ブラウザインタフェースの概要	29
▼管理コンソールにログインする	30
Portal Server 管理タグライブラリおよびポートレットの使用	30
psadmin コマンド行インタフェースの使用	31
2 ポータルおよび Portal Server インスタンスの管理	33
複数のポータルの概要	33
ポータルの設定	34
▼ポータルの一覧を表示する	35
▼ポータルを作成する	35
▼ポータルを削除する	36
▼ポータルデータをエクスポートする	37
▼ポータルデータをポータルにインポートする	38
Portal Server インスタンスの設定	39
▼ Portal Server インスタンスの一覧を表示する	39
▼ Portal Server インスタンスを作成する	40
▼ Portal Server インスタンスを削除する	41
3 組織、ロール、およびユーザーの管理	43
Portal Server での Access Manager の使用方法について	44

Portal Server の新しい組織の作成	45
▼ Portal Server で使用する新しい組織を作成する	45
▼ 新しい組織にアクセスする	45
Portal サービスの組織への追加	46
▼ Portal サービスを組織に追加する	46
▼ 新しいユーザーに必要な Portal サービスを指定する	47
特定のノードへの移動	49
ロケーションバーについて	49
▼ 新しいディレクトリノードを設定する	49
▼ ディレクトリノードをロケーションバーの選択肢に追加する	50
▼ ディレクトリノードをロケーションバーの選択肢から削除する	51
▼ ディレクトリノードの情報を表示する	51
4 Portal Server デスクトップの管理	53
Portal Server デスクトップ管理の概要	53
ディスプレイプロファイルの概要	53
デスクトップ属性の概要	55
Portal Server デスクトップコンテンツの管理	56
ポートレットの管理	56
▼ ポートレットを配備する	56
▼ ポートレットの配備を解除する	57
▼ ポートレットの設定を変更する	58
チャンネルおよびコンテナの管理	58
チャンネルおよびコンテナの表示	59
▼ DP XML ツリーおよびデスクトップ表示を表示する	60
チャンネルおよびコンテナのプロパティーの変更	61
▼ プロパティーを作成する	63
▼ リストを編集する	63
▼ チャンネルおよびコンテナのプロパティーを変更する	64
チャンネルおよびコンテナの作成と削除	65
▼ チャンネルまたはコンテナを作成する	65
▼ チャンネルまたはコンテナを削除する	66
タブの作成	67
▼ タブを作成する	67
チャンネルおよびコンテナの表示	67

▼チャンネルとコンテナをデスクトップに表示する	68
デスクトップ属性の管理	68
▼デスクトップ属性を設定する	69
ディスプレイプロファイルの管理	71
▼ディスプレイプロファイルをダウンロードする	72
▼ディスプレイプロファイルをアップロードする	72
▼ディスプレイプロファイルを削除する	73
5 リモートポートレットの Web サービス	75
WSRP 標準の概要	75
プロデューサの管理	76
登録をサポートするプロデューサの作成	76
▼登録をサポートするプロデューサを作成する	77
登録をサポートしないプロデューサの作成	78
▼登録をサポートしないプロデューサを作成する	78
WSRP プロデューサプロパティの有効化と編集	79
▼プロデューサのプロパティを有効化および編集する	79
登録検証クラスのカスタマイズ	80
登録ハンドルの生成	80
▼登録ハンドルを生成する	80
ebXML レジストリへのプロデューサ詳細情報の発行	81
▼Sun Java System Portal Server for Registry を設定する	81
▼プロデューサの詳細情報をレジストリに発行する	82
プロデューサの検索	83
▼プロデューサを検索する	83
コンシューマの管理	85
設定済みプロデューサの追加	85
▼設定済みプロデューサを追加する	86
ID 伝播機構	87
ダイジェストパスワードの設定	88
▼ダイジェストパスワードの受け入れを設定する	88
Web サービス SSO ポートレットを使用するユーザートークンプロファイルの作成	89
▼Web サービス SSO ポートレットを使用するユーザー資格を提供する	89
サービスの説明の更新	89

▼ サービスの説明を更新する	90
ユーザーカテゴリとロールのマッピング	90
▼ ポートレットでロールを作成する	91
▼ ユーザーカテゴリをロールにマッピングする	92
コンシューマ属性のマッピング	92
プロキシの設定	92
▼ 共通エージェントコンテナ内のコンシューマのプロキシを設定する	92
▼ Web コンテナ XML ファイルを設定する	93
WSRP プロデューサの管理	93
▼ WSRP プロデューサを作成する	93
▼ WSRP プロデューサを編集する	94
▼ コンシューマ登録を作成する	95
▼ コンシューマ登録を編集する	96
WSRP コンシューマの管理	97
▼ 設定済みプロデューサを追加する	97
▼ 設定済みプロデューサを編集する	97
▼ コンシューマ名を指定する	98
6 Portal Server エンドユーザー動作の追跡の管理	99
Portal Server ユーザー動作の追跡の概要	99
Portal Server ユーザー動作の追跡のセットアップ	101
▼ ユーザー動作の追跡ログを有効にする	101
▼ ユーザー動作の追跡レポートを生成する	102
7 Portal Server の動作の監視	103
Portal Server 監視について	103
Portal Server 監視の設定	104
▼ ポータル監視を有効または無効にする	104
▼ デスクトップ統計を表示する	104
▼ チャネル統計を表示する	105
Portal Server 監視データの収集	105
デスクトップ統計	105
チャネル統計	106

8 Portal Server のログインの管理	107
Portal Server のログインについて	107
Portal Server のログインの管理	107
▼ ログビューアを管理する	108
▼ ログ表示をカスタマイズする	109
▼ 共通ロガーの設定を管理する	109
▼ 個別ロガーの設定を管理する	111
9 Portal Server 登録の管理	115
Portal Server 登録の概要	115
登録のセットアップ	116
▼ 登録をセットアップする	116
Portal Server ディスカッションの管理	121
DiscussionProvider の概要	122
DiscussionProvider の管理	123
▼ DiscussionProvider からチャンネルを作成する	123
▼ DiscussionProvider チャンネルを削除する	124
▼ DiscussionProvider チャンネルを設定する	125
DiscussionLite チャンネル	125
10 Portal Server シングルサインオンアダプタの管理	127
シングルサインオンアダプタの概要	127
メタアダプタの管理	128
▼ メタアダプタを表示する	129
▼ メタアダプタを作成する	129
▼ アダプタを表示する	129
アダプタの管理	130
▼ アダプタを作成する	130
▼ アダプタの設定プロパティを編集する	131
匿名ユーザーの作成	131
▼ 匿名ユーザーのリストを作成する	131

パートII 検索サーバーの管理	133
11 検索サーバーの管理	135
検索サーバーの概要	135
検索データベース	136
データベース分類カテゴリ	136
検索サーバーの管理	137
▼検索サーバーを作成する	137
▼検索サーバーを削除する	138
データベースの概要	138
データベースへのインポート	138
データベーススキーマの編集	139
スキーマのエイリアスの定義	140
データベース分析の表示	140
データベースインデックスの再作成	140
データベースの期限切れ	140
データベースの破棄	141
データベースのパーティションの作成	141
データベースの管理	142
▼データベースを作成する	142
▼インポートエージェントを作成する	142
▼リソース記述を作成する	143
▼リソース記述を管理する	143
レポートの管理	144
▼レポートを表示する	144
カテゴリの管理	144
▼カテゴリを作成する	144
▼カテゴリを編集する	145
▼自動分類を実行する	145
▼自動分類の属性を編集する	145
12 検索サーバーロボットの管理	147
検索サーバーロボットについて	147
ロボットの動作の仕組み	148
ロボット設定ファイル	149

サイトの定義	150
ロボットの巡回の制御	150
ロボットユーティリティーの使用	151
ロボットのスケジュール設定	152
ロボットの管理	152
▼ロボットを起動する	152
▼ロボットデータベースをクリアーする	153
▼サイト定義を作成する	153
▼サイト定義を編集する	153
▼ロボットの巡回とインデックス作成を制御する	154
▼シミュレータを実行する	154
▼サイトプローブユーティリティーを実行する	155
リソースフィルタリング処理	155
フィルタ処理の段階	156
フィルタの構文	157
フィルタディレクティブ	158
フィルタの記述または修正	159
フィルタの管理	159
▼フィルタを作成する	159
▼フィルタを削除する	160
▼フィルタを編集する	160
▼フィルタを有効または無効にする	161
分類ルールの管理	161
▼分類ルールを作成する	161
▼分類ルールを編集する	162
ソースとデスティネーション	162
セットアップ段階で使用可能なソース	162
メタデータフィルタリング段階で使用可能なソース	163
データ段階で使用可能なソース	163
列挙、生成、およびシャットダウン段階で使用可能なソース	164
プロパティーの有効化	164
セットアップ関数	165
filterrules-setup	165
setup-regex-cache	165
setup-type-by-extension	166
フィルタリング関数	166

filter-by-exact	166
filter-by-max	167
filter-by-md5	167
filter-by-prefix	168
filter-by-regex	168
filterrules-process	169
サポート関数のフィルタリング	169
assign-source	169
assign-type-by-extension	170
clear-source	170
convert-to-html	170
copy-attribute	171
generate-by-exact	172
generate-by-prefix	172
generate-by-regex	173
generate-md5	173
generate-rd-expires	173
generate-rd-last-modified	174
rename-attribute	174
列挙関数	175
enumerate-urls	175
enumerate-urls-from-text	175
生成関数	176
extract-full-text	176
extract-html-meta	176
extract-html-text	177
extract-html-toc	177
extract-source	177
harvest-summarizer	178
シャットダウン関数	178
filterrules-shutdown	178
変更可能なプロパティー	179
robot.confファイルの例	186

パート III	委任管理の管理	187
13	委任管理チャンネルの管理	189
	ポータルの委任管理の概要	189
	委任管理チャンネルの設定	190
	▼ 委任管理チャンネルを設定する	190
14	Portal Server 委任管理タグライブラリの使用	197
	委任管理タグライブラリの概要	197
	▼ 委任管理タグのリファレンスを使用する	197
	索引	199

目次

図 12-1	ロボットの動作の仕組み	149
--------	-------------------	-----

表目次

表 6-1	ユーザー動作の追跡レポート	100
表 12-1	共通のメタデータタイプ	156
表 12-2	メタデータ段階でRAFで使用できるソース	163
表 12-3	データ段階でRAFで使用できるソース	164
表 12-4	ユーザーが変更できるプロパティ	179

例目次

例 12-1	列挙ファイルの構文	158
--------	-----------------	-----

はじめに

『Sun Java™ System Portal Server 7.1 管理ガイド』は、Sun Java System Portal Server 7.1 を管理するための情報と方法を説明します。

対象読者

このマニュアルは、Sun Java System サーバーとソフトウェアを使用してポータルサーバーの管理を担当する IT 管理者を対象にしています。

お読みになる前に

読者は、次の製品と概念について精通している必要があります。

- Sun Java System Directory Server
- Sun Java System Access Manager
- お使いの Web コンテナ
 - Sun Java System Application Server 8.2
 - Sun Java System Web Server 7.0
 - BEA WebLogic Server 8.1SP4
 - IBM WebSphere Application Server 5.1.1.6
- お使いのオペレーティングシステム
- 基本的な UNIX® の管理手順
- LDAP (lightweight directory access protocol)
- Web Services for Remote Portlets (WSRP)

内容の紹介

このマニュアルの章は3つの部分で構成されています。

- **パート I 「Sun Java System Portal Server の管理」**
 - 第 1 章は、Portal Server の管理方法の概要を説明しています。
 - 第 2 章は、Portal Server の設定と管理について説明しています。Portal Server のインスタンスの作成および削除の手順も含まれています。
 - 第 3 章は、組織とユーザーを管理する方法と、LDAP ノードを使用する方法を説明しています。
 - 第 4 章は、Portal Server を使用して提供されるエンドユーザーコンテンツの設定手順を説明しています。
 - 第 5 章は、Web Services for Remote Portlets (WSRP) を使用するための情報と手順を説明しています。
 - 第 6 章は、エンドユーザーの動作とエンドユーザーとさまざまなポータルシステムコンポーネントとの対話に関連する問題の診断、トラブルシューティング、分析の方法について説明します。
 - 第 7 章はデスクトップおよび Sun Java System Secure Remote Access サーバーについてのランタイム情報の入手方法について説明します。
 - 第 8 章は、Portal Server のロギングを管理する方法を説明します。
 - 第 9 章は、サブスクリプションの設定と管理方法について説明しています。
 - 第 10 章は、SSO アダプタの使用に関する情報を説明します。SSO アダプタは、ポータルへの認証された接続のためにこの設定データを提供します。SSO アダプタサービスはこのデータを保存します。
- **パート II 「検索サーバーの管理」**
 - 第 11 章は、検索カテゴリとデータベースでの作業についての詳細を説明します。
 - 第 12 章は、検索サーバーロボットと対応する設定ファイルについて説明します。
- **パート III 「委任管理の管理」**
 - 第 13 章は、管理機能の分散方法について説明します。
 - 第 14 章は、委任管理タグライブラリに使用できる参照情報について説明しています。

関連マニュアル

- 『Sun Java System Portal Server 7.1 配備計画ガイド』
- 『Sun Java System Portal Server 7 Technical Overview』
- 『Sun Java System Portal Server Secure Remote Access 7.1 Administration Guide』
- 『Sun Java System Portal Server 7.1 Command Line Reference』
- 『Tag Library for Delegated Administration』
- 『Sun Java System Portal Server 7.1 リリースノート (UNIX 版)』
- 『Sun Java System Portal Server 7.1 Community Sample Guide』
- 『Sun Java System Portal Server 7.1 Developer Sample Guide』
- 『Sun Java System Portal Server 7.1 Technical Reference』
- 『Sun Java System Portal Server 7.1 Developer's Guide』

Portal Server の概念とコンポーネントの概要については、『Sun Java System Portal Server 7 Technical Overview』を参照してください。

その他のサーバーマニュアル

その他のサーバーマニュアルについては、次を参照してください。

- Directory Server マニュアル (<http://docs.sun.com/coll/1224.1>)
- Access Manager マニュアル (<http://docs.sun.com/coll/1292.2>)
- Web Server マニュアル (<http://docs.sun.com/coll/1308.3>)
- Application Server マニュアル (<http://docs.sun.com/coll/1310.3>)
- Web Proxy Server マニュアル (<http://docs.sun.com/coll/1311.4>)

Sun 製品のマニュアルの検索

docs.sun.com Web サイトで Sun 製品のマニュアルを検索するだけでなく、検索フィールドに次の構文を入力して検索エンジンを使用できます。

search-term **site:docs.sun.com**

たとえば、「broker」を検索する場合は、次のように入力します。

broker site:docs.sun.com

検索に他の Sun Web サイト (java.sun.com、www.sun.com、developers.sun.com など) を含める場合は、検索フィールドの「**docs.sun.com**」の場所に「**sun.com**」と入力します。

関連サードパーティー Web サイトの参照

このマニュアルで参照しているサードパーティーの URL は追加の関連情報を提供します。

注-Sun は、このマニュアルに記載されているサードパーティーの Web サイトの有効性については責任を持ちません。Sun はそのようなサイトやリソースを通じて利用可能なコンテンツ、広告、製品、その他資料について、保証、責任、義務を負いません。Sun はそのようなサイトやリソースと通じて利用可能なコンテンツ、製品、サービスを、使用または信用したことによって、またはそれに関連して発生または申し立てられた、実際または疑惑のある損傷や損害に対しては責任または義務を負いません。

マニュアル、サポート、およびトレーニング

Sun のサービス	URL	内容
マニュアル	http://jp.sun.com/documentation/	PDF 文書および HTML 文書をダウンロードできます。
サポートおよび トレーニング	http://jp.sun.com/supporttraining/	技術サポート、パッチのダウンロード、および Sun のトレーニングコース情報を提供します。

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 <code>machine_name% you have mail.</code>

表 P-1 表記上の規則 (続き)

字体または記号	意味	例
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	<code>machine_name% su</code> <code>Password:</code>
<i>aabbcc123</i>	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、 <code>rm filename</code> と入力します。
『』	参照する書名を示します。	『コードマネージャー・ユーザーズガイド』を参照してください。
「」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第5章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	<code>sun% grep '^#define \ XV_VERSION_STRING'</code>

コード例は次のように表示されます。

- C シェル

```
machine_name% command y|n [filename]
```

- C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

- Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

- Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ(-)は2つのキーを同時に押すことを示します。たとえば、Ctrl-DはControl キーを押したままDキーを押すことを意味します。

パート I

Sun Java System Portal Server の管理

- 第1章
- 第2章
- 第3章
- 第4章
- 第5章
- 第6章
- 第7章
- 第8章
- 第9章
- 第10章

Portal Server 管理の概要

Portal Server の管理者は、次の項目に関するタスクを中心に、さまざまな機能を管理します。

- 複数のポータルおよび Portal Server インスタンス
- デスクトップ
- 検索サーバー
- Secure Remote Access サーバー
- シングルサインオン (SSO) アダプタ

この章では、Portal Server コンポーネント、およびポータルの管理方法について説明します。

- 27 ページの「Portal Server コンポーネントの概要」
- 28 ページの「Portal Server 管理コンソールの使用」
- 30 ページの「Portal Server 管理タグライブラリおよびポートレットの使用」
- 31 ページの「psadmin コマンド行インタフェースの使用」

Portal Server コンポーネントの概要

Portal Server の配備には、ポータルの管理に影響を及ぼす多数のコンポーネントが含まれています。主なコンポーネントは次のとおりです。

- 共通エージェントコンテナ - Java 管理アプリケーションのコンテナを実装するスタンドアロン Java プログラム。詳しくは、『Solaris 10 What's New』を参照してください。
- ポータル管理サーバー - Portal Server MBean にアクセスするユーザーの認証およびアクセス制御チェックを実行する管理アプリケーション。このサーバーは、JMX™ インタフェースを使用し、共通エージェントコンテナモジュールとして実装されます。ポータル管理サーバーインスタンスは、Portal Server 製品がインストールされているホストごとに実行されます。

- ポータルドメインリポジトリ - Portal Server MBean の編成方法についての情報が格納される階層型データストア。このリポジトリには、一部の Portal Server MBean の設定データも保存されます。デフォルトのポータルドメインリポジトリは、Access Manager で使用される同じ LDAP サーバー内のサブツリーです。
スタンドアロンゲートウェイのインストールで、ゲートウェイから LDAP サーバーと通信することはできません。ローカルゲートウェイ MBean 情報のみを格納するために、ゲートウェイファイルシステム上の追加のポータルドメインリポジトリが使用されます。
- ポータルデータストア - リレーショナルデータベース管理システム (RDBMS) や LDAP サーバーなどのバックエンド記憶領域またはファイルシステム内で、ポータルによるコンテンツ配信を容易にする設定データ、その他の Portal Server リソースが格納される場所。
- ポータル管理 MBean - ポータル管理サーバーによって共通エージェントコンテナサーバーにロードされ、ポータル管理タスクを実行します。
- ポータル管理コマンド行インタフェース (psadmin) - 各 Portal Server コンポーネントの管理ツールを提供します。詳しくは、31 ページの「psadmin コマンド行インタフェースの使用」を参照してください。
- ポータル管理コンソール (psconsole) - 各種のポータルサーバーリソースを管理するためのブラウザインタフェースです。詳しくは、28 ページの「Portal Server 管理コンソールの使用」を参照してください。
- 監視 MBean - Portal Server の実行時リソース情報の収集をサポートします。詳しくは、第7章「Portal Server の動作の監視」を参照してください。
- ローカルファイルシステムデータ - ローカルファイルシステムに格納されるポータルデータ。ポータルデータには、各種の設定ファイル、プロバイダベースのテンプレートと JSP™ 構文ファイル、リソースバンドルファイル、カスタマイズされたプロバイダベースの Java クラスなどがあります。

Portal Server コンポーネントの詳細については、『Sun Java™ System Portal Server 7.1 配備計画ガイド』を参照してください。

Portal Server 管理コンソールの使用

さまざまなポータル管理タスクを容易にする Portal Server 管理コンソールは、次のような機能を持つ Java™ 2 Platform, Enterprise Edition (J2EE™) アプリケーションです。

- Web ブラウザで使用できる
- 設定したデバッグレベルに従ってメッセージをデバッグログに記録する
- 名前と値のペアを含む設定の変更を記録する
- JMX (Java Management Extensions) テクノロジーを使用して、ポータル管理サーバー内のポータル管理 MBean と通信し、ポータルデータストアに接続する

ポータル管理者が管理コンソールを使用して実行できるタスクは、次のとおりです。

- デスクトップおよびコンテンツ配信を管理する
- ユーザーの操作を追跡し、ポータル管理者がその情報を利用して、エンドユーザーの活動に関連した問題、およびエンドユーザーと各 Portal Server コンポーネントとの対話状況を診断、トラブルシューティング、および分析する
- Portal Server のデスクトップおよび Secure Remote Access コンポーネントに関連する実行時統計を取得する
- Portal Server アプリケーションに関する情報を記録する

ブラウザインタフェースの概要

管理コンソールのユーザーインタフェースでは、管理機能がページごとに分類されています。各ページの上部にはタブの列が表示されます。各タブは、分類された管理機能が収められているページを表します。タブをクリックすると、ページを切り替えることができます。表示されるタブは次のとおりです。

- 共通タスク - ポータル管理者が頻繁に実行するタスクを直接起動できるリンクが表示される
- ポータル - ポータル管理者が特定のポータルを選択できるように、配備済みのポータルが Portal ID の順に一覧表示される
- 検索サーバー - 特定の検索サーバーを管理するページにポータル管理者がアクセスできるように、特定の検索サーバーの名前が一覧表示される
- **Secure Remote Access** - ポータル管理者は、リモートユーザーがインターネットを経由してポータルおよびそのサービスを安全に利用する方法を管理できる
- **SSO アダプター** - ポータル管理者は、一度サインインしたエンドユーザーがアプリケーションの認証済みアクセス権を取得する方法を管理できる

Portal Server の管理者は、特定のエンドユーザーのアイデンティティを定義して、ポータルのコンテンツの使用を許可および制限することができます。ポータルコンテンツを特定のエンティティで使用可能にするため、ポータルページ、属性、およびアクセスポリシーを設定できます。このようなエンティティは次のとおりです。

- 特定の組織
- 特定のサブ組織
- ロール
- 個々のエンドユーザー

▼ 管理コンソールにログインする

SuperAdmin アクセス権を持つ管理者のみが、Portal Server 管理コンソールを使用できます。ユーザーは、特別の URI (Uniform Resource Identifier) からブラウザクライアントを使用して Portal Server 管理コンソールにアクセスします。

- 1 ブラウザで次の URL を入力します:**http:// hostname:port /psconsole**
hostname 管理コンソールを実行しているシステムの名前。
port インストール時に割り当てられた管理コンソールのポート番号。
- 2 テキストボックスに管理ユーザー名とパスワードを入力します。
管理ユーザーは、最上位管理者にしてください。通常、管理ユーザー名は `amadmin` です。
- 3 「ログイン」 ボタンをクリックします。
管理コンソールの「共通タスク」 ページが表示されます。

Portal Server 管理タグライブラリおよびポートレットの使用

Portal Server の管理タグライブラリを使用して、管理コンソールの代わりにデスクトップからポータルを管理できるようにする、管理ポートレットを開発できます。このタグライブラリを使用して管理者が実行できるタスクは次のとおりです。

- 事前に設定された管理ポートレットの変更
- 新しい管理機能を持つポートレットの開発
- ユーザー管理タスク、プロバイダ管理タスク、およびポートレットと WSRP 管理タスクのサポート
- JSPProvider に基づくチャンネルの作成と管理
- カスタムユーザーインタフェースを備えたカスタム管理ポートレットの記述
- 任意のカスタムチャンネルを管理する管理ポートレットの記述

管理者は、管理ポートレットを使用して、委任管理者と呼ばれるほかのユーザーに委任管理ステータスを付与することができます。Portal Server には、委任管理者用の基本デスクトップの設計に使用できる管理ポートレットのサンプルセットがあります。

詳しくは、『Sun Java System Portal Server 7.1 Developer Sample Guide』および『Tag Library for Delegated Administration』を参照してください。

psadmin コマンド行インタフェースの使用

Portal Server ソフトウェアでは、コマンド行インタフェース (CLI) を使用できます。ポータル管理者は CLI で次の操作が可能です。

- キーボードからのコマンド入力による管理タスクの実行
- 定期的に繰り返す管理タスクのスクリプトへの組み込みによる自動化

CLI には、ポータルタスクを管理するための多数の psadmin サブコマンドがあります。主なサブコマンドの機能は次のとおりです。

- 複数のポータルおよびポータルインスタンスの管理
- ポータルおよびポートレット WAR ファイルの配備
- 検索サーバーの管理
- Secure Remote Access サーバーの管理
- 監視の管理
- ポータルログの管理

ほとんどのサブコマンドは、ブラウザインタフェースの対応する機能と同じ感覚で使えるようになっています。特別のコマンドがない管理機能については、標準 UNIX コマンドを使用します。



注意 - Portal Server を Sun Java System Web Server 上にインストールした場合は、psadmin コマンドを呼び出す前に Web Server の管理サーバーを起動する必要があります。

すべての psadmin サブコマンドについては、『Sun Java System Portal Server 7.1 Command Line Reference』を参照してください。

ポータルおよび Portal Server インスタンスの管理

この章では、複数のポータルと、ポータルおよび Portal Server インスタンスの管理方法について説明します。この章で説明する内容は次のとおりです。

- 33 ページの「複数のポータルの概要」
- 34 ページの「ポータルの設定」
- 39 ページの「Portal Server インスタンスの設定」

複数のポータルの概要

複数のポータルは、同じユーザーセットを共有します。複数のポータルの主な機能は次のとおりです。

- ポータルが URL で識別される。たとえば、次のようになります。
`http://hr.xyz.com/portal` または `http://eng.xyz.com/portal`
- 同じユーザーリポジトリ、つまり同じ Access Manager およびディレクトリサーバーを共有する。Access Manager を使用してエンドユーザーを管理するので、LDAP のエンドユーザーデータをどのリポジトリとも同期する必要がありません。エンドユーザーに関連するすべてのデータが、ただ1つのディレクトリサーバーで管理されます。
- 複数のポータルと Portal Server インスタンスを1つ以上のホストに配備できる。たとえば、1つのホスト上に、1つのポータルのコンテンツを配信する2つの Portal Server インスタンスと、別のポータルを配信する3つの Portal Server インスタンスを配備することができます。各 Portal Server インスタンスは、別個の Web コンテナインスタンス内で実行する必要があります。

次のコンポーネントはすべてのポータルで共有されます。

- リライタ - このコンポーネントは共有されますが、ポータルごとに異なるルールセットを定義できます。
- SSO アダプタ - このコンポーネントは共有されますが、ポータルごとに異なるアダプタを定義できます。

- Secure Remote Access のすべてのサービス

次のコンポーネントはポータルと1対1に対応します。

- デスクトップ - 各ポータルには独立したデスクトップがあります。
- 登録 - ポータルごとに別個に設定されます。
- WSRP - プロデューサとコンシューマ - プロデューサおよび設定済みのプロデューサのセットで、ポータルごとに独立しています。

検索とポータルは、多対多の関係に設定できます。

- 1つのポータルで1つの検索サーバーを使用できる。
- 複数のポータルで1つの検索サーバーを使用できる。
- 各ポータルで複数の検索サーバーを使用できる。

エンドユーザーに表示されるコンテンツはポータルごとに異なり、各ポータルのデスクトップはエンドユーザーがカスタマイズできます。ポータル間でシングルサインオンが可能です。たとえば、会社で2つのポータルを使用できるエンドユーザーは、多くの場合、次のように操作します。

- Portal 1 の URL を入力し、会社のアイデンティティを使用して認証を行います。
- Portal 1 のパーソナライズされたコンテンツを利用します。
- Portal 2 の URL を入力しますが、認証を行う必要はありません。
- Portal 2 のパーソナライズされたコンテンツを利用します。

ポータルが2つ以上あっても使用する Access Manager が異なる場合は、複数のポータルとはいいません。それらは独立した互いに無関係なポータルで、独自のユーザーセットを持っています。

Access Manger は、固有のインスタンスのコレクションとして実行できます。その場合、すべてのインスタンスで同じ Directory Server インスタンスのセットが使用されます。異なる Access Manager とは、互いに無関係な2つの Access Manager のことで、同じ Access Manager の異なるインスタンスのことではありません。

ポータルの設定

ポータルは、同一のコンテンツを配信し、単一の URL (Uniform Resource Locator) にマップされる1つ以上の Portal Server インスタンスから成ります。ポータルから配信されるコンテンツとサービスは、ポータルのすべてのインスタンスで共通です。

複数のポータルは、同じユーザーセットを共有します。それらのポータルは1つ以上のホストに配備できますが、すべてのポータルで同じユーザーリポジトリ、つまり同じ Access Manager およびディレクトリサーバーを共有します。

注-ポータルが2つ以上あっても使用する Access Manager が異なる場合は、複数のポータルとはいいません。それらは独立した互いに無関係なポータルで、独自のユーザーセットを持っています。

Access Manger は、固有のインスタンスのコレクションとして実行できます。その場合、すべてのインスタンスで同じ Directory Server インスタンスのセットが使用されます。異なる Access Manager とは、互いに無関係な2つの Access Manager のことで、同じ Access Manager の異なるインスタンスのことではありません。

ここでは、次のタスクの実行方法について説明します。

- 35 ページの「ポータルの一覧を表示する」
- 35 ページの「ポータルを作成する」
- 36 ページの「ポータルを削除する」
- 37 ページの「ポータルデータをエクスポートする」
- 38 ページの「ポータルデータをポータルにインポートする」

▼ ポータルの一覧を表示する

設定済みの Portal Server の一覧を表示できます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。

参考 同等の psadmin コマンド

```
psadmin list-portals
```

▼ ポータルを作成する

Portal Server のインストール時に、*portal1* という名前のデフォルトポータルが作成されます。「新規ポータル」ウィザードを使用して新しいポータルサーバーを作成することもできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「新規ポータル」ボタンをクリックしてウィザードを起動します。
- 4 作成する **Portal Server** に **portal5** など、一意の名前を指定します。

- 5 エンドユーザーが **Portal Server** にアクセスできるように、`/portal` などの **URI** を入力します。
- 6 **Web** コンテナタイプを選択します。
使用できるタイプは次のとおりです。
 - Sun Java™ System Web Server 6.0
 - Sun Java System Web Server 7.x
 - Sun Java System Application Server 8.x
 - BEA WebLogic 8.1SP4/SP5
 - IBM WebSphere 5.1.1.6
- 7 (省略可能) デフォルト **Web** コンテナインスタンスのプロパティを変更します。
詳細は、『Sun Java System Portal Server 7.1 Configuration Guide』の「Creating a New Portal」を参照してください。
- 8 入力した情報を確認します。
- 9 「完了」をクリックすると、新しいポータルが作成されます。
- 10 (省略可能) 処理を監視するにはログファイルを表示します。
 - a. ポータルが作成されるマシンにログインします。
 - b. `psadmin set-logger` コマンドを実行します。
`./psadmin set-logger -u uid -f password -m component-type -O logger-name`

参考 同等の psadmin コマンド

```
psadmin create-portal
```

サポートされている Web コンテナの `webcontainer.properties` のテンプレートは、`portal-install-dir/template` ディレクトリにあります。

▼ ポータルを削除する

すべてのホスト上にあるポータルの既存インスタンスをすべて削除して、Access Manager LDAP ディレクトリ内のそのポータルのデータをクリーンアップすることができます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。

- 3 削除するポータルをリストから選択して、「ポータルを削除」ボタンをクリックします。

参考 同等の psadmin コマンド

```
psadmin delete-portal
```

▼ ポータルデータをエクスポートする

par ファイルに保存できるポータルデータは次のとおりです。

- Access Manager ディレクトリに保存されているデータ
- デスクトップファイルシステムファイル。デフォルトの場所は `/var/opt/SUNWportal/portals/portal-id/desktop` ディレクトリです
- デスクトップカスタマイズクラス。デフォルトの場所は `/var/opt/SUNWportal/portals/portal-id/desktop/classes` ディレクトリです
- Portal Server Web アプリケーション。デフォルトの場所は `/var/opt/SUNWportal/portals/portal-id/war` ディレクトリです
- Portal Server Web ソースデータ。デフォルトの場所は `/var/opt/SUNWportal/portals/portal-id/web-src` ディレクトリです

アーカイブに保存したデータは、同じポータルまたは別のポータルにインポートできます。psconsole からポータルをエクスポートするには、次の手順に従います。

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 テーブルからポータルを選択します。
- 4 「エクスポート」ボタンをクリックします。
- 5 **Portal Server マシン上の par ファイルの場所、およびエクスポートするデータを指定します。**
 - すべてのデスクトップデータ — エクスポート先の par に、ファイルシステムデータおよびディスプレイプロファイルデータが保存されます
 - ファイルシステムデータのみ — エクスポート先の par ファイルに、ポータルデスクトップおよびポータル web-src に配備されるデータ、つまりデスクトップファイルシステムデータのみが保存されます

- ディスプレイプロファイルデータのみ — エクスポート先の par に、ディスプレイプロファイルデータのみが保存されます

参考 同等の psadmin コマンド

```
psadmin export
```

注 - このコマンドでは、Directory Server 内のユーザーデータはサポートされません。

▼ ポータルデータをポータルにインポートする

エクスポート済みの任意のポータルデータを任意のポータルにインポートできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 テーブルからポータルを選択します。
「デスクトップデータのインポート」ページが表示されます。
- 4 「インポート」ボタンをクリックし、次の情報を指定します。
 - インポートするデータの par ファイルパス。par ファイルは、**Portal Server** システム上に存在する必要があります。
 - ポータルのストレージ構造とインポートするアーカイブファイルとが一致しない場合に、続行するかどうか。
- 5 ポータルの **Web** アプリケーションを再配備します。
 - a. psadmin redeploy コマンドを実行する時刻をスケジュールします。
この手順は、休憩時間中に行うか、システムが本稼働していないシステム保守モードで実行するように計画します。この処理により、ポータル war ファイルが再配備され、デスクトップを実行中のユーザーはログアウトされるため、ユーザーの作業内容が失われます。
 - b. psadmin redeploy コマンドを実行します。

```
psadmin redeploy -u amadmin -f passwordfile -p portallID --allwebapps
```

参考 同等の psadmin コマンド

```
psadmin import
```

注 - このコマンドでは、Directory Server 内のユーザーデータはサポートされません。

Portal Server インスタンスの設定

Portal Server インスタンスは、Web コンテナに配備された Web アプリケーションです。インスタンスは、固有の Portal Server コンテキスト URI を使用して、特定のネットワークポートで要求を処理します。各 Portal Server インスタンスは単一のポータルと関連付けられます。

サーバーインスタンスは固有のポート上で待機し、1つの IP アドレスまたはホストの任意の IP アドレスのいずれかにバインドされます。Portal Server の場合、サーバーインスタンスは、ポート上で待機し単一の Java™ 仮想マシン (JVM™ ソフトウェア) を実行する、配備コンテナプロセスに対応します。

注 - Sun Java™ System Web Server と Sun Java™ System Application Server は複数のインスタンスをサポートします。

ここでは、次のタスクの実行方法について説明します。

- 39 ページの「Portal Server インスタンスの一覧を表示する」
- 40 ページの「Portal Server インスタンスを作成する」
- 41 ページの「Portal Server インスタンスを削除する」

▼ Portal Server インスタンスの一覧を表示する

設定済みの Portal Server インスタンスの一覧を表示できます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 テーブルから **Portal Server** の名前をクリックします。
- 4 「サーバーインスタンス」タブを選択します。
選択した Portal Server のすべてのインスタンスが一覧表示されます。

参考 同等の psadmin コマンド

```
psadmin list-portals
```

▼ Portal Server インスタンスを作成する

- 始める前に
- Web コンテナインスタンス上の既存の Portal Server の新規インスタンスを作成します。
 - Web コンテナインスタンスを起動します。
 - Web コンテナの管理サーバーを起動します。
- 1 **Portal Server** 管理コンソールにログインします。
 - 2 「ポータル」タブを選択します。
 - 3 **Portal Server** の名前を選択します。
 - 4 「サーバーインスタンス」タブを選択します。
 - 5 「新規インスタンス」ボタンをクリックしてウィザードを起動します。
 - 6 **Portal ID** の名前を指定します。
 - 7 **Web** コンテナタイプを選択します。
使用できるタイプは次のとおりです。
 - Sun Java System Web Server 7
 - Sun Java System Application Server 8.2
 - BEA WebLogic 8.1SP4
 - IBM WebSphere 5.1.1.6
 - 8 (省略可能) デフォルト **Web** コンテナインスタンスのプロパティを変更します。
詳細は、『Sun Java System Portal Server 7.1 Configuration Guide』の「Creating a Portal on the Same Node」を参照してください。
 - 9 入力した情報を確認し、新しいポータルインスタンスを作成する場合は「完了」をクリックします。
進捗バーにこの手順の進行状況が表示されます。手順が完了すると、結果ページが表示されます。
 - 10 「完了」をクリックすると、新しいポータルインスタンスが作成されます。

参考 同等の psadmin コマンド

```
psadmin create-instance
```

▼ Portal Server インスタンスを削除する

Portal Server のインスタンスを削除することができます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 **Portal Server** の名前を選択します。
- 4 「サーバーインスタンス」タブを選択します。
- 5 削除するインスタンスをテーブルから選択します。
- 6 「インスタンスを削除」ボタンをクリックします。

参考 同等の psadmin コマンド

```
psadmin delete-instance
```


組織、ロール、およびユーザーの管理

Portal Server 管理者は、特定のエンドユーザーの ID の定義によって、ポータル上のコンテンツへのアクセスを、許可および制限できます。特定のエンティティがポータルコンテンツを使用できるように、ポータルページ、属性、アクセスポリシーを設定できます。これらのエンティティには、以下が含まれます。

- 特定の組織
- 特定のサブ組織
- ロール
- 個別のエンドユーザー

組織、ロール、およびエンドユーザーを管理するために、Portal Server の管理者は、Portal Server 管理コンソールと Sun Java™ System Access Manager コンソールの両方を使用する必要があります。この章では、Portal Server 管理者が Access Manager を使用して行う場合の方法について説明します。この章では、次のトピックについて説明します。

- 44 ページの「Portal Server での Access Manager の使用方法について」
- 45 ページの「Portal Server の新しい組織の作成」
- 49 ページの「特定のノードへの移動」

注- この章では、旧バージョンモードをサポートするようインストールされ、設定された Access Manager の使用方法について説明します。旧バージョンモードとレルムモードについては、『Sun Java System Access Manager 管理ガイド』を参照してください。

Portal Server での Access Manager の使用方法について

Portal Server は、Sun Java System Access Manager サービスを使用して、Portal Server のエンドユーザーとアプリケーションに固有の属性を管理します。ID に関連するタスクの管理には、Access Manager コンソールを使用する必要があります。

ポータルサイトにアクセスできる人物を制御するために、Portal Server 管理者は次のツールを使用する必要があります。

- Portal Server 管理コンソールは、管理者が以下について管理できるブラウザインタフェースです。
 - ポータルとポータルインスタンス
 - 検索
 - リモートアクセス
 - シングルサインオン
 - ディスプレイプロファイルドキュメント
 - コンテナとチャンネル
- Sun Java System Access Manager コンソールは、さまざまなレベルのアクセス権限を持つ管理者が以下を実行できるブラウザインタフェースです。
 - レルムと組織の作成と削除
 - これらの組織のユーザーの作成と削除
 - サービスの管理
 - 組織リソースへのアクセスを保護および制限する強制ポリシーの設定

Portal Server 管理者は、Access Manager を使用して、次のタスクを実行する必要があります。

- 組織またはサブ組織内でのロールに基づいてユーザーに適切なアクセス権限を割り当て、管理するために、ユーザー、ロール、組織を含む ID ベースのオブジェクトを管理する
- エンドユーザーを認証することによって、特定のエンドユーザーに管理機能を委任し、組織、サブ組織、ユーザー、ポリシー、ロール、およびチャンネルを管理する

Access Manager は、LDAP (lightweight directory access protocol) を使用します。

Access Manager の管理については、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。

Portal Server の新しい組織の作成

新しい組織は、最上位レベルの Access Manager 組織で登録されたサービスを継承します。新しい組織の継承する一般的なサービスは、次のとおりです。

- Access Manager の設定
 - 認証の設定
- 認証モジュール
 - コア
 - LDAP
 - ポリシー設定

新しい組織は、LDAP 認証を使用し、LDAP サービス設定は、対応するグローバルサービスから継承されます。

Access Manager の管理については、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。

▼ Portal Server で使用する新しい組織を作成する

- 1 Access Manager コンソールにログインします。

Access Manager の管理については、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。

- 2 「アイデンティティ管理」で「表示」メニューから「組織」を選択します。
- 3 「新規」をクリックして新しい組織を作成します。

- 4 組織属性を指定します。
たとえば、次のようになります。

名前 TestOrganization

組織エイリアス TestOrganization

- 5 「了解」をクリックします。

▼ 新しい組織にアクセスする

- ブラウザに次の URL を入力します。

http:// host:port/amserver/UI/Login?org= organizationalias

host コンソールが実行されているシステムの名前。

port	インストール中に割り当てられたコンソールのポート番号。
organizationalias	「組織エイリアス」属性フィールドに割り当てられた値。

Portal サービスの組織への追加

Portal にアクセスできるようになる前に、組織にいくつかのサービスを追加する必要があります。組織に追加する必要のあるサービスには、次のようなものがあります。

- Portal Server 設定
 - portalID デスクトップ
 - portalID 登録
 - SSO アダプタ
 - portalID WSRP コンシューマ
- モバイルアプリケーションの設定
 - モバイルアドレス帳
 - モバイルカレンダー
 - モバイルメール

追加できるオプションサービスには次のようなものがあります。

- SRA 設定
 - アクセス権一覧
 - NetFile
 - ネットレット
 - プロキシレット

▼ Portal サービスを組織に追加する

組織が Portal Server にアクセスできるようになる前に、組織にいくつかのサービスを追加する必要があります。組織に Portal サービスを追加した後、Portal Server 管理コンソールを使用して、Portal Server 設定を管理します。

- 1 **Access Manager** コンソールにログインします。
Access Manager の管理については、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。
- 2 「アイデンティティ管理」で「表示」メニューから「組織」を選択します。

- 3 自分の組織をクリックします。
たとえば、次のようになります。TestOrganization
- 4 組織の「表示」メニューで「サービス」を選択します。
- 5 「追加」をクリックします。
- 6 配備で可能な場合は次のサービスを選択します。
 - モバイルアプリケーションの設定
 - モバイルアドレス帳
 - モバイルカレンダー
 - モバイルFAX
 - モバイルメール
 - Portal Server 設定
 - portalID デスクトップ
 - portalID 登録
 - SSO アダプタ
 - リモートポートレット (WSRP)
 - portalID WSRP コンシューマ
 - SRA 設定
 - アクセス権一覧
 - NetFile
 - ネットレット
 - プロキシレット
- 7 「了解」をクリックします。

▼ 新しいユーザーに必要な Portal サービスを指定する

組織に Portal サービスをすべて追加した後、Access Manager コンソールを使用して、新しく作成したエンドユーザーが Portal Desktop と必要な Portal サービスにアクセスできるように、サービスを追加する必要があります。

Access Manager 管理サービスでは、作成時にエンドユーザーエントリに動的に追加されるサービスを指定できます。「Sign-Me Up」機能など Portal 配備でユーザーが作成できる場合は、組織の Access Manager コンソールで「必要なサービス」設定を指定します。

始める前に Portal サービスを組織に追加します。46 ページの「Portal サービスの組織への追加」を参照してください。

- 1 **Access Manager** コンソールにログインします。
Access Manager の管理については、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。
- 2 管理サービスを追加します。
 - a. 「アイデンティティ管理」で「表示」メニューから「組織」を選択します。
 - b. 自分の組織をクリックします。
たとえば、次のようになります。TestOrganization
 - c. 組織の「表示」メニューで「サービス」を選択します。
 - d. 「追加」をクリックします。
 - e. 「管理」を選択して「了解」をクリックします。
- 3 「管理」の必須サービスの設定を指定します。
この設定によって、必要なサービスリストのサービスをすべて新しいエンドユーザーに割り当てるかどうかを指定します。
 - a. サービスから「管理」を選択します。
 - b. 「必要なサービス」設定で次のサービスを指定します。
 - SunPortalportalIDDesktopService
 - SunPortalportalIDSubscriptionsService
 - SunMobileAppABService
 - SunMobileAppCalendarService
 - SunMobileAppMailService
 - SunSSOAdapterService
 - c. 「保存」をクリックします。
- 4 **Access Manager** コンソールからログアウトします。

特定のノードへの移動

Portal Server は Access Manager サービスを使用して、アプリケーションとユーザー固有の属性を保存します。LDAP ディレクトリノード (DN) のポータル関連の機能を管理できるようにするため、Portal Server 管理コンソールでは、DN についての詳細がタブの行の下にある水平の帯であるロケーションバーに表示されます。

ロケーションバーを使用すると、次の動作を行うことができます。

- 現在選択しているノードを特定する
- 組織 DN を最高 10 個まで表示する
- 別のディレクトリ名に変更する

ディレクトリ名は、組織、ロール、またはユーザーの名前にできます。

ロケーションバーについて

ロケーションバーには次の機能があります。

- **DN を選択** - このドロップダウンメニューを使用して、次のディレクトリノードタイプを表示できます。
 - Portal Server のインストール時に定義されたデフォルト組織。
 - 「DN の追加」 ボタンを使用して管理者が設定したノード。
- **選択した** - 現在選択されている DN を特定します。
- **DN の入力** - 完全名を入力してすでに定義されている DN に移動できます。

▼ 新しいディレクトリノードを設定する

ロケーションバーに追加せずに新しい DN を選択できます。

- 1 **Portal Server 管理コンソールにログインします。**
- 2 ロケーションバーの隣にある「追加」 ボタンを選択します。
- 3 次のいずれかの方法を使用して **DN** の名前を選択します。
 - ウィンドウに一覧表示された **DN** を選択する。
 - 「検索」ユーティリティを使用する。
 - a. 検索文字列を入力します。
ワイルドカード文字を使用できます。

検索結果は、省略名と対応するディレクトリノードで表示されます。

b. 「検索」 ボタンをクリックします。

4 「現在のDNを設定」 ボタンをクリックします。

ウィンドウが閉じ、「選択した」 フィールドに新しいディレクトリノードが表示されます。ディレクトリノードは、ロケーションバーの選択肢に追加されません。

▼ ディレクトリノードをロケーションバーの選択肢に追加する

ロケーションバーのメニューにディレクトリノードを追加すると、cookie として保存されるため、ディレクトリノードはセッションが終わるまで同じブラウザで使用できます。

1 **Portal Server** 管理コンソールにログインします。

2 次のいずれかの方法を使用して **DN** の名前を選択します。

■ 「DNの追加」 ボタンを使用する。

a. 「DNを選択」 メニューの隣にある「追加」 ボタンをクリックします。

「DNリストに追加」 ポップアップウィンドウが開き、使用可能なディレクトリノードのリストが表示されます。

b. 目的の **DN** を選択します。

■ 「検索」 ユーティリティを使用する。

a. 「検索」 メニューを使用してオブジェクトタイプを選択します。

b. 検索文字列を入力します。

ワイルドカード文字を使用できます。

検索結果は、省略名と対応する **DN** で表示されます。

c. 目的の **DN** を選択します。

3 ディレクトリノードの名前を選択します。

4 (省略可能) 省略名フィールドを編集して、ドロップダウンメニューに表示されるディレクトリノードの名前を変更します。

- 5 「追加」 ボタンをクリックします。
ディレクトリノードが「DN を選択」メニューに追加されます。

▼ **ディレクトリノードをロケーションバーの選択肢から削除する**
ロケーションバーに表示されるドロップダウンリストからディレクトリノードを削除できます。ディレクトリノード自体は削除されません。ディレクトリ名をLDAPデータベースから削除するには、Access Manager を使用する必要があります。

インストール中に定義されたデフォルトの組織は削除できません。

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「DN を選択」ドロップダウンメニューから削除する DN を選択します。
- 3 「DN を選択」ドロップダウンメニューボタンの隣にある「削除」ボタンをクリックします。
選択したディレクトリノードが削除されます。

▼ **ディレクトリノードの情報を表示する**

- 1 **Portal Server 管理コンソールにログインします。**
- 2 次のいずれかの方法で、ディレクトリノードについての情報を表示します。
 - 「DN の入力」テキストボックスにディレクトリノードの名前を入力し、「ジャンプ」ボタンをクリックする。
 - 「DN を選択」メニューからディレクトリノードの名前を選択する。

Portal Server デスクトップの管理

この章では、Sun Java™ System Portal Server デスクトップとその管理方法について説明します。

- 53 ページの「Portal Server デスクトップ管理の概要」
- 56 ページの「Portal Server デスクトップコンテンツの管理」
- 68 ページの「デスクトップ属性の管理」
- 71 ページの「ディスプレイプロファイルの管理」

Portal Server デスクトップ管理の概要

ここでは、Portal Server デスクトップの主要コンポーネントについて説明します。説明する内容は次のとおりです。

- 53 ページの「ディスプレイプロファイルの概要」
- 55 ページの「デスクトップ属性の概要」

ディスプレイプロファイルの概要

Portal Server をインストールするときに、最初の組織を作成します。その際、指定した入力パラメータに基づいて、ディスプレイプロファイルグローバルレベルのドキュメントおよびデフォルト組織のディスプレイプロファイルがインポートされます。

その後は、組織、サブ組織、またはロールを新規に作成しても、ディスプレイプロファイルは毎回自動的にロードされません。ただし、それらの新しい組織、サブ組織、またはロールでは、親から定義済みのディスプレイプロファイルが継承されます。新規に作成した組織、サブ組織、またはロールに固有のエントリがある場合は、ディスプレイプロファイルを手動でロードする必要があります。

ディスプレイプロファイルでは、次の3つの項目を定義して標準デスクトップの表示設定を作成します。

プロバイダ定義

プロバイダの名前と Java クラスを指定します。プロバイダは、コンテンツの生成に使用されるテンプレートで、チャンネルに表示されます。

チャンネル定義

プロバイダクラスのインスタンスの実行時設定を指定します。チャンネルはコンテンツの単位で、多くの場合は行と列で編成されます。また、チャンネル内にチャンネルを持つことができます。これは、コンテナチャンネルと呼ばれます。

プロバイダとチャンネルのプロパティ定義

プロバイダとチャンネルのプロパティに値を指定します。プロバイダに定義されたプロパティは、通常はそのプロバイダに関連するチャンネルの、デフォルトの値を指定します。チャンネルの表示設定には、タイトル、説明、チャンネル幅などのプロパティが含まれます。チャンネルに定義するプロパティでは、通常、デフォルト値とは異なるそのチャンネル固有の値を指定します。

コンテナプロパティでは、コンテナに含まれるチャンネルの表示方法に関する表示定義を定義します。これには、コンテナのレイアウト(狭い-広い、広い-狭い、または狭い-広い-狭い)、コンテナ内のチャンネルのリスト、チャンネルの位置(行と列の番号)、コンテナ内のチャンネルのウィンドウ状態(最小または切り離し)などがあります。

ディスプレイプロファイルはチャンネルのプロパティ値のみを提供します。ユーザーのデスクトップに実際に表示されるコンテンツの全体的なレイアウトや構造を定義するものではありません。ただし、ディスプレイプロファイルは、テーブルコンテナの列レイアウトやテーブルコンテナがテーブル内にチャンネルを描画する方法など、チャンネル表示のいくつかの側面を間接的に制御します。

無効な XML を含むディスプレイプロファイルドキュメントを保存しようとする、システムがエラーを出力します。このエラーメッセージは、タイトル、メッセージ、およびサブメッセージとして表示されます。メッセージボックスのタイトルは、「無効な XML ドキュメント」です。メッセージは、次のいずれかの形式で表示されます。

- XML のパースに失敗しました。
- XML の doctype がありません。
- DP の保存に失敗しました。
- 無効な XML 入力です。

「無効な XML ドキュメント」エラーが表示された場合、エラーを修正して XML ドキュメントを保存可能にする必要があります。

ドキュメントを表示する構文は次のとおりです。

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<DOCTYPE DisplayProfile SYSTEM " jar://resources/psdp.dtd">
```

```
<DisplayProfile version="1.0" priority="xxx">
  <Properties>
    ...
  </Properties>
  <Channels>
    ...
  </Channels>
  <Providers>
    ...
  </Providers>
</DisplayProfile>
```

デスクトップ属性の概要

デスクトップでは、ユーザーのディスプレイプロファイルのマージしたセットに含まれるすべてのドキュメントをさらにマージし、それを使用してユーザーのデスクトップを構成します。ディスプレイプロファイルのマージしたセットには、ユーザーに関連するすべてのディスプレイプロファイルドキュメントが含まれます。ディスプレイプロファイルは、Portal Server 組織ツリーの異なるレベルで定義されます。ツリーのさまざまなレベルのディスプレイプロファイルドキュメントをマージまたは結合して、ユーザーのディスプレイプロファイルを作成します。

たとえば、ユーザーのディスプレイプロファイルドキュメントをロールディスプレイプロファイルドキュメント (存在する場合)、組織のディスプレイプロファイルドキュメント、およびグローバルディスプレイプロファイルドキュメントとマージして、ユーザーのディスプレイプロファイルを形成します。

デスクトップディスプレイプロファイルとその他の設定データは、Sun Java System Access Manager サービス管理フレームワークで、ポータルデスクトップサービスの親コンテナ、デスクトップタイプ、編集コンテナなどのサービス属性として定義されます。組織が Sun Java System Access Manager 管理コンソールでポータルデスクトップサービスを追加すると、組織内のすべてのユーザーが、自分のユーザープロファイルでポータルデスクトップサービス属性を継承します。これらの属性はポータルデスクトップで照会され、情報を集約してポータルデスクトップに表示する方法が決まります。

68 ページの「[デスクトップ属性の管理](#)」を参照してください。

Portal Server デスクトップコンテンツの管理

ここでは、デスクトップコンテンツの管理方法について説明します。デスクトップの詳細については、『Sun Java System Portal Server 7 Technical Overview』の「Understanding the Standard Desktop」を参照してください。

- [56 ページの「ポートレットの管理」](#)
- [58 ページの「チャンネルおよびコンテナの管理」](#)

ポートレットの管理

ここでは、ポートレットの配備方法と配備解除方法、およびポートレットの設定の変更方法について説明します。

ポートレットは、ポータルコンテキストの範囲内で要求を処理し、コンテンツを生成する Web アプリケーションです。ポートレットは、ポートレットコンテナ (JSR 168 エキスパートグループによって定義されるポートレット仕様の実装) によって管理されます。

ポートレットは、選択した DN ノードに一度しか配備できません。同一の DN ノードにポートレットがすでに配備されている場合は、そのポートレットの配備を解除してから配備してください。1つのポートレットを複数のサブ組織またはロールに配備する必要がある場合は、ポートレットをポータルのグローバル DN または親組織に配備します。

- [56 ページの「ポートレットを配備する」](#)
- [57 ページの「ポートレットの配備を解除する」](#)
- [58 ページの「ポートレットの設定を変更する」](#)

▼ ポートレットを配備する

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 「ポートレットを配備」をクリックしてウィザードを起動します。
 - a. 選択したポータルと選択した DN がポートレットの配備先と一致することを確認し、「次へ」をクリックします。

- b. ポートレット **war** ファイル、ロールファイル、およびユーザーファイルを指定します。

注-ロールファイルとユーザーファイルは省略できます。war ファイル、ロールファイル、およびユーザーファイルは、ローカルマシン、またはリモートポータルサーバーシステムのいずれかに配置できます。

- c. ローカルシステムかリモートポータルサーバーシステムのいずれかのボタンを選択します。
- ローカルマシンからファイルをアップロードする場合は、参照ダイアログボックスを使用してローカルマシンからファイルを選択します。
 - リモートポータルサーバーシステムからファイルをアップロードする場合は、ファイルチューザダイアログを使用してリモートマシンからファイルを選択します。
- d. 指定した情報を確認し、「次へ」をクリックします。
- e. ポートレットが配備される際に情報ページが表示されます。
- 6 画面の指示に従ってポートレットを配備します。

参考 同等の psadmin コマンド

```
psadmin deploy-portlet
```

▼ ポートレットの配備を解除する

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 「ポートレットを配備解除」をクリックしてウィザードを起動します。
- 6 必要に応じて設定属性を変更します。
- 7 「配備解除」をクリックして変更を記録します。

参考 同等の psadmin コマンド

```
psadmin undeploy-portlet
```

▼ ポートレットの設定を変更する

- 1 **Portal Server 管理コンソール**にログインします。
- 2 「共通タスク」タブをクリックし、サブメニューから「コンテナとチャンネルを管理」を選択します。
- 3 ポートレットを配備するポータルおよび **DN** を選択します。
使用できるチャンネルとポートレットがナビゲーションツリーに表示されます。
- 4 左のフレームのナビゲーションツリーからポートレットチャンネルを選択します。
右のフレームには、設定テーブルとプロパティテーブルが表示されます。
- 5 設定テーブルで、変更する設定の「値を編集」リンクをクリックします。
- 6 設定ウィザードでテキストフィールドに新しい値を入力し、「了解」をクリックします。
 - 値を削除するには、リストからその値を選択して「削除」をクリックします。
- 7 設定の変更が終わったら「保存」をクリックします。
- 8 「閉じる」をクリックします。

チャンネルおよびコンテナの管理

ここでは、ポータルサーバーのチャンネルとコンテナを管理コンソールから管理する方法について説明します。

この章で説明する内容は次のとおりです。

- 59 ページの「チャンネルおよびコンテナの表示」
- 61 ページの「チャンネルおよびコンテナのプロパティの変更」
- 65 ページの「チャンネルおよびコンテナの作成と削除」
- 67 ページの「タブの作成」
- 67 ページの「チャンネルおよびコンテナの表示」

チャンネルおよびコンテナの表示

ユーザー用のデスクトップは、デスクトップ親コンテナを起動することで描画されます。親コンテナ属性は、すべての組織、ロール、およびユーザー DN でカスタマイズできます。特定の DN でのデスクトップのコンテンツは、デスクトップ親コンテナ内で表示するように選択した子コンテナとチャンネルの反復によって生成されます。

通常、デスクトップ親コンテナには、少数のタブコンテナまたはテーブルコンテナが含まれます。親コンテナの選択済みノードのリストにあるタブコンテナごとに、1つのタブがユーザーデスクトップ上に表示されます。タブの下に表示されるチャンネルが、タブコンテナ内のチャンネルです。

ポータル管理コンソールの「チャンネルおよびコンテナ管理」の左下フレームには、次の2つのコンポーネントがあります。

- 「表示タイプ」メニュー
- チャンネルおよびコンテナツリー

「表示タイプ」メニューの項目およびツリーに表示されるノードは、マージされたディスプレイプロファイル XML のコンテンツに依存しています。

ツリーにはコンテナおよびチャンネルノードが含まれます。コンテンツをデスクトップに配信するチャンネルには、次の3つのタイプがあります。

- プロバイダ(ネイティブ)チャンネル
- ポートレットチャンネル
- リモートポートレットチャンネル

ツリー内のノードリンクをクリックして、右のフレームにプロパティーと操作を表示できます。

「表示タイプ」メニューの項目には、次の2つのタイプがあります。

- DP XML ツリー
- デスクトップ表示

60 ページの「[DP XML ツリーおよびデスクトップ表示を表示する](#)」を参照してください。

DP XML ツリー

このツリーには、マージされたディスプレイプロファイル (DP) XML にあるチャンネルとコンテナの完全なセットが表示されます。DP XML ツリーのルート要素は DP_ROOT で、ディスプレイプロファイルのすべてのチャンネルおよびコンテナの親になります。チャンネルは DP_ROOT の直下にも、DP_ROOT の下のコンテナ内にも作成することができます。

DP XML ツリーに一覧表示されているノードが常にデスクトップに表示されるとは限りません。ディスプレイプロファイル内には、デスクトップコンテナの階層で参照されたり、階層に含まれたりすることがまったくないノードもあります。

たとえば、デスクトップデフォルトコンテナ `JSPTabContainer` には2つのコンテナ、`tab1` と `tab2` があります。`tab1` に `ch1` と `ch2` が含まれ、`tab2` に `ch3` と `ch4` が含まれている場合、DP XML ツリーには5つのチャンネルが定義されます。DP XML ツリーでは、コンテナ階層内の `ch1` から `ch4` が参照されますが、`ch5` は参照されません。そのため、デスクトップに表示されるのは、`ch1` から `ch4` だけです。

デスクトップ表示

デスクトップ表示は、マージされたディスプレイプロファイルで使用可能な最上位コンテナです。各デスクトップ表示は、DNにあるデスクトップの親コンテナとして設定できます。デスクトップ表示を選択すると、コンテンツをデスクトップに描画する役割を持つチャンネルとコンテナの階層がツリーに表示されます。

デスクトップ表示で表示されるチャンネルとコンテナには、次の2つの状態があります。

- デスクトップで選択され表示可能
- 選択可能

注-この状態の場合、チャンネルとコンテナのアイコンがグレーで表示されます。

右のフレームにあるタスクリンクをクリックして、デスクトップ表示内のチャンネルおよびコンテナの状態を変更することができます。コンテナまたはチャンネルアイコンの上にマウスを移動すると、状態のツールチップが表示されます。ツールチップでノードの完全修飾名も確認することができます。

▼ DP XML ツリーおよびデスクトップ表示を表示する

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択し、「DNを選択」ドロップダウンメニューから任意のDNを選択します。
 - 「コンテナとチャンネルの管理」ページの「DNを選択」メニューから組織を選択することもできます。
- 4 「タスク」で「コンテナとチャンネルの管理」をクリックします。

- 5 「表示タイプ」ドロップダウンメニューから「DPXML ツリー」または「デスクトップ表示」を選択します。

チャンネルおよびコンテナのプロパティの変更

ここでは、チャンネルとコンテナのプロパティ、およびその変更方法について説明します。

実行できるタスクは次のとおりです。

- 63 ページの「プロパティを作成する」
- 63 ページの「リストを編集する」
- 58 ページの「ポートレットの設定を変更する」
- 64 ページの「チャンネルおよびコンテナのプロパティを変更する」
- 72 ページの「ディスプレイプロファイルをアップロードする」

プロパティの概要

ツリー内のノードをクリックして表示されるプロパティは、最上位のプロパティまたはチャンネルレベルのプロパティです。これらのプロパティはプロバイダレベルで定義され、チャンネルに合わせてカスタマイズできます。ただし、チャンネルに追加された新しいプロパティをプロバイダに追加することはできません。チャンネルレベルで新規プロパティを追加できないのは、このためです。

プロパティテーブルには、クライアントタイプとロケールが示されます。プロパティのタイプを示す列はありませんが、次の規約が適用されます。

文字列	「値」列が、最大 30 文字の広いテキストフィールドになります。
整数	「値」列が、最大 5 文字の狭いテキストフィールドになります。
ブール	「値」がラジオボタンになります。
マップ	「名前」がリンクになります。
リスト	「値」列が「値を編集」リンクになります。このリンクをクリックすると、値を追加および削除するウィザードが起動します。
空のコレクション	名前が「値の編集」リンクを表示するリンクになります。空のコレクションに名前と値のペアを追加して、対応表のように機能させることができます。追加すると「値の編集」が表示されなくなります。「値の編集」ウィザードを使用して空のコレクションに値を追加すると、コレクションはリストとして動作するので、名前のリンクは表示されません。

プロパティテーブルには、「名前」と「値」列に加えて2つの列があります。

カテゴリ プロパティが拡張か基本かを示します。拡張プロパティは、経験を積んだ管理者向けです。

状態 どのプロパティも次の3つの状態のいずれかになります。

- デフォルト - プロバイダで値が割り当てられています。
- 継承 - 上位のいずれかのレベルで値が変更されています。たとえば、現在のノードがロールの場合、プロパティはロールの組織でカスタマイズされたと考えられます。この組織は、親組織または親組織の親の可能性があります。プロパティが継承される場合はリンクとなります。このリンクをクリックすると、このプロパティの継承元からの階層にある、可能性のあるすべての親ノードが表示されます。
- カスタマイズ - このノードで値が定義されました。

プロパティテーブルには次のボタンがあります。

カスタマイズを削除 このノードで定義された値をディスプレイプロファイルから削除します。このとき、階層内のいずれかの親でプロパティがカスタマイズされていると、プロパティがそこから継承される場合があります。階層内のどの場所でも値がカスタマイズされていない場合、プロバイダで定義された値が表示され、状態が「デフォルト」として示されません。

保存 値の追加、削除、および変更を保存します。

リセット 変更を無視して、データストアから最後に保存された状態に値をリセットします。

すべてのソートを解除 すべてのソートを解除します。

ヒント-テーブルは、任意の列タイトルをクリックしてソートすることができます。「名前」ボタンを最初にクリックして名前でソートすると、「カテゴリ」ボタンと「状態」ボタンの横に+が表示されます。この+をクリックすると、次のソート条件が適用されます。

テーブルの設定 テーブルの設定を変更します。

変更しないかぎり、クライアントタイプおよびロケールはデフォルトに設定されます。

▼ プロパティを作成する

「新規プロパティ」ウィザードから、値の編集と保存を行うことができます。名前と値の新しいペアを追加することも可能です。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「作業」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからコンテナを選択して、右のフレームに「プロパティの編集」ページを表示します。
- 7 「新規プロパティ」ボタンをクリックしてウィザードを起動します。
- 8 プロパティタイプを選択し、「次へ」をクリックします。
- 9 「名前」を入力し、「値」を選択し、拡張プロパティかどうかを指定します。

注-名前と値のペアを含むコレクションプロパティは、マップのように機能します。コレクションタイプのプロパティは入れ子にできます。テーブルの上のプロパティパスは現在の入れ子のレベルに応じて変化し、入れ子の外側に戻ることが可能です。

末尾に付加される値は省略できます。たとえば、enやen_USは値として指定できますが、usのみを指定することはできません。ロケールを指定する標準Java形式が適用されます。

- 10 「完了」をクリックしてプロパティを作成します。
- 11 「閉じる」をクリックすると、テーブルに新しいプロパティが表示されます。

▼ リストを編集する

値のみを含むコレクションプロパティは、リストのように機能します。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。

- 3 「ポータル」からポータルを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「作業」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからコンテナを選択して、右のフレームに「プロパティの編集」ページを表示します。
- 7 プロパティの「値を編集」リンクをクリックしてウィザードを起動します。
- 8 必要な変更を行います。
 - 値を追加する場合は、「新しい値」テキストボックスに値の名前を入力し、「追加」をクリックします。
 - 値を削除する場合は、「値」リストから値を選択し、「削除」をクリックします。
- 9 「閉じる」をクリックします。
「プロパティの編集」ページのリスト内の値の数が更新されます。

▼ チャンネルおよびコンテナのプロパティを変更する

- 1 [Portal Server 管理コンソールにログイン](#)します。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「作業」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからチャンネルまたはコンテナを選択して、右のフレームに「プロパティの編集」ページを表示します。
- 7 プロパティを変更して「保存」をクリックします。

参考 同等の psadmin コマンド

```
psadmin modify-dp
```

チャンネルおよびコンテナの作成と削除

ここでは、チャンネルとコンテナをポータル管理コンソールから作成および削除する方法について説明します。

- 65 ページの「チャンネルまたはコンテナを作成する」
- 66 ページの「チャンネルまたはコンテナを削除する」

▼ チャンネルまたはコンテナを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「作業」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからコンテナを選択して、右のフレームに「プロパティの編集」ページを表示します。
- 7 「作業」で「新規チャンネルまたはコンテナ」をクリックして、ウィザードを起動します。
ウィザードで、選択したポータルと選択したDNがチャンネルまたはコンテナの作成先と一致することを確認し、「次へ」をクリックします。
- 8 ウィザードからコンテナまたはチャンネルを作成します。
 - コンテナを作成するには次の手順を実行します。
 - a. 「コンテナプロバイダ」ドロップダウンメニューからプロバイダを選択し、「次へ」をクリックします。
 - b. 「チャンネルまたはコンテナ名」テキストフィールドに名前を入力し、「次へ」をクリックします。
 - c. 選択内容を確認して「完了」をクリックします。
コンテナの作成を確認するメッセージが表示されます。
 - d. 「閉じる」をクリックします。

- チャンネルを作成するには次の手順を実行します。
 - a. チャンネルタイプを選択します。
次の3つのタイプからチャンネルを選択します。
 - 「プロバイダチャンネル」を選択すると、プロバイダチャンネルのリストが表示されます。
 - 「JSR 168 ポートレットチャンネル」を選択すると、ポートレットチャンネルのリストが表示されます。
 - 「WSRP リモートポートレットチャンネル」を選択する場合は、ドロップダウンメニューから登録済みプロデューサとリモートポートレットを選択します。
 - b. 「チャンネルまたはコンテナ名」テキストフィールドに名前を入力し、「次へ」をクリックします。
 - c. 選択内容を確認して「完了」をクリックします。
チャンネルの作成を確認するメッセージが表示されます。
 - d. 「閉じる」をクリックします。

参考 同等の psadmin コマンド

```
psadmin add-dp
```

▼ チャンネルまたはコンテナを削除する

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 「タスク」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからコンテナを選択して、右のフレームに「プロパティの編集」ページを表示します。
- 7 「タスク」で「削除するチャンネルまたはコンテナを選択」をクリックします。

- 8 「タイプ」で「チャンネルまたはコンテナ」を選択します。
使用できるチャンネルおよびコンテナが表示されます。
- 9 チャンネルまたはコンテナを選択し、「削除」をクリックします。

参考 同等の psadmin コマンド

```
psadmin remove-dp
```

タブの作成

ここでは、Portal Server 管理コンソールからタブを作成する方法について説明します。

▼ タブを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 「タスク」で「コンテナとチャンネルを管理」をクリックします。
- 6 左のフレームのツリーからタブコンテナを選択します。
- 7 右のフレームの「タスク」で「新しいタブ」をクリックして、ウィザードを起動します。

チャンネルおよびコンテナの表示

ここでは、エンドユーザーデスクトップにチャンネルとコンテナを表示する方法について説明します。エンドユーザーがデスクトップに表示するチャンネルとコンテナを選択できるように、それらをコンテンツページ上で使用可能にすることもできます。

▼ チャンネルとコンテナをデスクトップに表示する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルを選択します。
- 4 「タスク」で「コンテナとチャンネルを管理」をクリックします。
- 5 左のフレームのツリーからコンテナを選択して、右のパネルに「プロパティの編集」ページを表示します。
- 6 「タスク」で「チャンネルおよびコンテナをポータルデスクトップ上に表示または非表示」をクリックします。
- 7 「使用可能」でチャンネルまたはコンテナを選択します。
- 8 「追加」ボタンを使用してチャンネルを移動し、コンテンツページまたはポータルデスクトップ上に表示されるようにします。
 - 「削除」ボタンを使用して、チャンネルまたはコンテナを「使用可能」に戻すことができます。
- 9 「保存」をクリックします。

参考 同等の `psadmin` コマンド

```
psadmin modify-dp.
```

デスクトップ属性の管理

ここでは、デスクトップ属性の管理方法について説明します。詳しくは、[55 ページ](#)の「[デスクトップ属性の概要](#)」を参照してください。

デスクトップ属性は、最上位の組織と組織ツリーの他のレベルとで異なります。ロケーションバーを「TopLevel」に変更すると、グローバルデスクトップ属性を表示できます。ほかの識別名を選択すると、組織またはロールのデスクトップ属性を表示できます。

▼ デスクトップ属性を設定する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択し、次に「デスクトップ」を選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「デスクトップ属性」で必要に応じて設定属性を変更します。

使用できるオプションは次のとおりです。

COS 優先順位

複数のデスクトップテンプレートをマージする際の競合の解決に使用する、デスクトップサービステンプレートの競合の解決レベルを設定します。この属性は組織とロールにのみ適用され、ユーザーとグローバルDNには適用されません。

親コンテナ

不特定のプロバイダでデスクトップが呼び出される場合に描画するデフォルトコンテナを指定します。「親コンテナ」の値には、ポータルページ上にヘッダーおよびフッターを描画可能な `TopLevelContainer` として定義されるいずれかのコンテナを指定できます。ディスプレイプロファイルのプロパティ `TopLevel` が `true` に設定されている場合、コンテナは最上位コンテナとなります。

編集コンテナ

URL で編集コンテナが指定されていない場合にコンテンツのラップに使用する、デフォルト編集コンテナを指定します。このコンテナは、チャンネルのタイトルバーで編集リンクがクリックされたときに、親コンテナによって編集ページの描画に使用されます。

デスクトップタイプ

テンプレートおよびJSPの検索時にデスクトップの検索操作で使用されるコマンドで区切ったリストです。検索はリストの最初の要素から開始されます。各要素は、デスクトップテンプレートベースディレクトリ下のサブディレクトリを表します。たとえば、"sampleportal,foo" というリストの場合、検索

デスクトップ属性	<p>は sampleportal ディレクトリ、foo ディレクトリ、デフォルトディレクトリの順番で行われます。</p> <p>ロールに関連付けられたユーザーにデスクトップ属性を表示するかどうかを指定します。この動的な属性は、管理タグライブラリで主にロールに基づく委任管理に使用されます。この属性の表示を有効にすると、委任管理者が、親組織から継承されたチャンネルとコンテナを管理できるようになります。この属性は、組織およびロールにのみ適用されません。</p>
ディスプレイプロファイル優先順位	<p>ディスプレイプロファイルドキュメントの優先順位を設定します。ディスプレイプロファイルドキュメントは、低い優先順位から高い優先順位の順にマージされます。低い数値は低い優先順位を表します。たとえば、1 は2 よりも優先順位が低くなります。マージセマンティクスの結果、優先順位が低いドキュメントで設定された値は、優先順位が高いドキュメントによって無効になります。ただし、優先順位が低いドキュメントでオブジェクトのマージがロックされている場合を除きます。</p>
<hr/> <p>注-ディスプレイプロファイル優先順位は、デスクトップサービス属性として保存されません。</p> <hr/>	
<p>次の属性は、「TopLevel [グローバル]」DN にのみ適用されます。</p>	
XML パース検証 連携	<p>XML 構文解析の検証を有効にします。</p> <p>アイデンティティ連携を有効にし、ユーザーが複数のインターネットサービスプロバイダ、ローカルアイデンティティを関連付け、接続、またはバインドできるようにします。これにより、ユーザーのネットワークアイデンティティを1つにできます。</p>
ホストプロバイダ ID	<p>ユーザーのネットワークアイデンティティを提供するホストの一意の識別子を指定します。</p>

セッションの取得間隔	セッションの取得間隔を秒単位で指定します。
セッションアイドル時間	セッションを終了するまでのアイドル時間を秒単位で指定します。
クライアントセッションの最大数	クライアントセッションの瞬間最大数を指定します。
匿名デスクトップ	有効にすると、選択したポータルで匿名デスクトップを使用できます。
連携ユーザーの匿名アクセス	ホストプロバイダ上にネットワークアイデンティティを持つユーザーが、ユーザー名とパスワードの入力によってポータルデスクトップにアクセスすることを禁止します。
匿名デスクトップで有効な UID	認証を受けずにデスクトップにアクセスする権限を持つユーザー ID のリスト。

- 6 「保存」をクリックして変更を記録します。
 そうしない場合は、「リセット」をクリックして編集を取り消します。

注-グローバル属性を変更する場合は、ロケーションバードロップダウンの DN を「TopLevel」に変更します。

参考 同等の psadmin コマンド

psadmin undeploy-portlet

ディスプレイプロファイルの管理

ここでは、Sun Java System Portal Server ディスプレイプロファイルの管理方法について説明します。詳しくは、53 ページの「ディスプレイプロファイルの概要」を参照してください。

ポータル管理コンソールから実行できるタスクは次のとおりです。

- 72 ページの「ディスプレイプロファイルをダウンロードする」
- 72 ページの「ディスプレイプロファイルをアップロードする」
- 73 ページの「ディスプレイプロファイルを削除する」

▼ ディスプレイプロファイルをダウンロードする

ディスプレイプロファイルをファイルにダウンロードできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「タスク」で「ディスプレイプロファイルをダウンロード」をクリックします。ブラウザのダウンロードポップアップウィンドウが開きます。
- 6 場所を選択して「保存」をクリックします。

注-この手順はブラウザによって異なる場合があります。

参考 同等の psadmin コマンド

```
psadmin get-attribute
```

▼ ディスプレイプロファイルをアップロードする

ディスプレイプロファイルをファイルにアップロードできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 5 「タスク」で「ディスプレイプロファイルをアップロード」をクリックします。
- 6 「参照」ボタンを使用して、アップロードするディスプレイプロファイルのファイルを選択します。

注- 目的のファイルは、ユーザーのブラウザの設定に従ってローカルマシン上にあります。

- 7 「アップロード」をクリックします。

参考 同等の psadmin コマンド

```
psadmin modify-dp.
```

▼ ディスプレイプロファイルを削除する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 「タスク」で「ディスプレイプロファイルの削除」をクリックします。
- 6 削除を確認するダイアログボックスで「了解」をクリックします。

参考 同等の psadmin コマンド

```
psadmin remove-dp
```


リモートポートレットの Web サービス

Sun Java™ System Portal Server はリモートポートレットの Web サービス (WSRP) をサポートします。この章では、WSRP を使用する際のガイドラインと最良の事例を示します。この章で説明する内容は次のとおりです。

- 75 ページの「WSRP 標準の概要」
- 76 ページの「プロデューサの管理」
- 85 ページの「コンシューマの管理」
- 93 ページの「WSRP プロデューサの管理」
- 97 ページの「WSRP コンシューマの管理」

WSRP 標準の概要

WSRP 1.0 は OASIS 標準の 1 つで、これによって、リモートアプリケーションおよびコンテンツのポータルへの統合が簡単にできます。WSRP 標準は、プレゼンテーション指向の対話式 Web サービスを、一般的な定義済みのインタフェースおよびプロトコルとともに定義して、ユーザー対話を処理し、プレゼンテーションを、ポータルによる仲介と集約、およびそれらのサービスを発行、検索、バインドするための規約に適した形にフラグメント化します。

WSRP インタフェースは一般的で定義済みであるため、WSRP 標準を実装するすべての Web サービスは WSRP 標準のすべてのポータルに接続できます。すなわち、あらゆる WSRP サービスを統合するには、サービスに依存しない単一のアダプタがポータル側にあれば十分です。したがって、WSRP は、コンテンツおよびアプリケーションプロバイダが、ポータルを実行している組織に対して複雑なプログラミング作業なく各自のサービスを提供するための手段です。

詳細については、WSRP 1.0 標準を参照してください。

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrp

Portal Server における WSRP 1.0 標準の実装には、WSRP のコンシューマとプロデューサの両方が含まれます。WSRP プロデューサの実装は、リモートの WSRP コン

シューマによって使用される JSR 168 ポートレットの発行をサポートします。JSR 168 ポートレットは、ポータルサーバー上にローカルに配備されています。これらのポートレットは、WSRP プロデューサのインスタンスによって発行できます。

もう1つのポータルサーバーは、WSRP コンシューマを介してこれらのリモートポートレットに登録できます。ローカルポートレットがポータルの基本機能の大部分を提供すると予想される一方で、リモートポートレットは、インストール作業、あるいはコンシューマ側のポータルサーバーでコードをローカルに実行する必要もなく、各種のリモートポートレットをバインドする可能性を提供します。

プロデューサの管理

ここでは次の作業について説明します。

- 76 ページの「登録をサポートするプロデューサの作成」
- 78 ページの「登録をサポートしないプロデューサの作成」
- 79 ページの「WSRP プロデューサプロパティの有効化と編集」
- 80 ページの「登録検証クラスのカスタマイズ」
- 80 ページの「登録ハンドルの生成」
- 81 ページの「ebXML レジストリへのプロデューサ詳細情報の発行」
- 83 ページの「プロデューサの検索」

ローカルに配備されたポートレットを、WSRP コンシューマとして動作する別のポータルヘリモートで提供する場合は、プロデューサを作成します。ポータルは複数のプロデューサをホストできます。コンシューマは、プロデューサが提供するリモートのポートレットをインポートできます。WSRP コンシューマに提供するポートレットをベースにして、1つ以上のプロデューサを作成できます。プロデューサは登録をサポートできますが、登録が必須であるわけではありません。プロデューサが登録をサポートする場合、コンシューマをプロデューサと組み合わせて動作させるには、コンシューマを登録する必要があります。

登録をサポートするプロデューサの作成

登録は、コンシューマとプロデューサの間の技術的または業務上の関係を構築するために使用されます。プロデューサの作成時には、次の登録機構のいずれかを定義できます。すなわち、in-band 登録と out-of-band 登録です。

プロデューサが登録を必要とし、in-band 登録を有効とした場合、コンシューマは WSRP インタフェースを介して詳細を提供し、プロデューサに登録できます。また、コンシューマには out-of-band 通信を介した登録のオプションも提供されます。すなわち、コンシューマは out-of-band 通信を介して取得される登録ハンドルを提供できます。

プロデューサが登録を必要とし、out-of-band 登録を有効とした場合、コンシューマは out-of-band 通信を介して登録ハンドルを取得し、登録処理中に登録ハンドルを提供します。out-of-band 登録は、電話の呼び出しや電子メールなど手動の介入を伴って発生します。out-of-band 登録をサポートするプロデューサは、out-of-band 通信を介してコンシューマに関する詳細を取得し、コンシューマの登録ハンドルを作成します。登録ハンドルは、out-of-band 通信を介してコンシューマに送られます。

▼ 登録をサポートするプロデューサを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」タブをクリックします。
- 5 「DN を選択」ドロップダウンメニューでいずれかの DN を選択し、「プロデューサ」タブをクリックします。
「WSRP プロデューサ」テーブルに作成済みのすべてのプロデューサが表示されます。

注 - Sun Java System Identity Server に組織が作成されます。ポートレットの利用度に基づいて組織またはサブ組織の DN を選択します。

- 6 「新規」をクリックして、新しいプロデューサを作成します。
- 7 プロデューサを識別する名前を入力します。
- 8 「登録」の「必須」を選択します。
- 9 **Sun Java System Portal Server** アプリケーションインタフェースを使用して設定済みプロデューサの追加中に、コンシューマが詳細を入力するには、in-band 登録の「サポートする」を選択します。
- 10 登録プロパティを追加するには、「行を追加」をクリックします。値を入力します。登録プロパティの名前と説明を入力します。

注 - 登録プロパティとは、コンシューマを特定のプロデューサに登録するときにコンシューマから取得される詳細情報です。コンシューマによって入力された登録プロパティは、登録検証クラスを介して検証可能です。

- 11 コンシューマが電話の呼び出しや電子メールなど **out-of-band** 通信を介して詳細情報を提供する場合、「**out-of-band** 登録」の「サポートする」を選択します。
- 12 「次へ」をクリックします。
「確認」画面に入力した詳細情報が表示されます。詳細情報を確認します。「戻る」をクリックすると、入力済みの詳細情報を変更できます。
- 13 「終了」をクリックします。

参考 同等の psadmin コマンド

```
psadmin create-producer
```

登録をサポートしないプロデューサの作成

登録を必要としないプロデューサに関しては、コンシューマは out-of-band 通信を介した情報の入力または取得を行う必要はありません。この場合、コンシューマはプロデューサによって提供されたポートレットをカスタマイズ (編集) することはできません。登録をサポートしないプロデューサは、読み取り専用のポータルをコンシューマに提供します。

▼ 登録をサポートしないプロデューサを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」タブをクリックします。
- 5 **DN**を選択します。
「設定済みのプロデューサ」テーブルに、すでに設定済みのすべてのプロデューサが表示されます。
- 6 「新規」をクリックします。
- 7 プロデューサの名前を入力します。
- 8 「登録」の「不要」を選択します。
- 9 「終了」をクリックします。

参考 同等の psadmin コマンド

```
psadmin create-producer
```

WSRP プロデューサプロパティの有効化と編集

新しく作成したプロデューサは、コンシューマが登録できるように有効化してください。1つ以上のポートレットを追加することでプロデューサを有効化できます。

プロデューサを無効にすることも可能です。ただし、無効にされたプロデューサに登録されたコンシューマはいずれも、そのプロデューサによって提供されるポートレットにアクセスできなくなります。

▼ プロデューサのプロパティを有効化および編集する

- 1 「プロデューサ」タブでプロデューサ名のリンクをクリックします。
「プロパティの編集」画面が表示されます。この画面には WSDL (Web Services Definition Language) URL が表示されます。WSDL URL は、特定のプロデューサに固有の URL であり、コンシューマがプロデューサにアクセスする際に使用します。
- 2 1つ以上の発行済みポートレットをプロデューサに追加します。

注-プロデューサは少なくとも1つの発行済みポートレットを有効にする必要があります。画面には、プロデューサが作成されているポータルに関連付けられた発行済みポートレットがすべて表示されます。

- 3 ポートレットを選択し、「追加」をクリックします。
- 4 必要に応じて「登録検証クラス」フィールドを編集します。
Registration Validator が、コンシューマによって入力された登録プロパティの検証に使用されます。必要に応じて、このクラスをカスタマイズすることも可能です。
- 5 「保存」をクリックします。これで、画面上の「有効」チェックボックスを編集できます。「有効」を選択し、「保存」をクリックします。

注-プロデューサのその他のプロパティを編集することもできます。

参考 同等の psadmin コマンド

```
psadmin set-attribute
```

登録検証クラスのカスタマイズ

登録検証クラスはカスタマイズ可能です。このクラスを使用すると、登録プロパティを扱うことができます。たとえば、顧客の郵便番号の検証などです。登録検証は、WSRP プロデューサの登録検証に使用する SPI です。検証クラスのカスタマイズの詳細については、<http://portalID/portal/javadocs/desktop> を参照してください。また、WSRP:Validating Registration Data を『Sun Java System Portal Server 7.1 Developer's Guide』で参照することもできます。

登録ハンドルの生成

登録をサポートするプロデューサでは、特定のコンシューマに対して登録ハンドルを生成する必要があります。登録ハンドルの生成後、out-of-band 通信を介してプロデューサに登録するためにコンシューマと通信する必要があります。プロデューサへの登録時に、コンシューマは登録ハンドルを入力する必要があります。

▼ 登録ハンドルを生成する

- 1 「コンシューマ登録」タブをクリックします。

特定のプロデューサにすでに登録されているすべてのコンシューマが画面に表示されます。

- 2 「新規」をクリックします。

- 3 名前、状態、コンシューマエージェント、メソッドなど詳細情報を入力します。

コンシューマ名	コンシューマを識別する一意の名前。
状態	有効または無効にすることができます。
コンシューマエージェント	コンシューマのベンダーの名前とバージョンを指定します。コンシューマエージェント名は <code>ProductName.MajorVersion.MinorVersion</code> の形式にしてください。ProductName はコンシューマが該当の配備にインストールした製品を識別します。majorVersion と minorVersion はベンダーが定義した製品バージョンの表示です。—この文字列には、製品やコンシューマが提供する任意の文字や単語を追加できません。
メソッド	コンシューマが、get メソッドを持つフォームを含む HTML マークアップをサポートする形式でポートレット URL を実装しているかどうかを指定します。

- 4 「次へ」をクリックします。
プロデューサの作成時に指定する登録プロパティの値が画面に表示されます。
- 5 値を入力し、「次へ」をクリックします。「終了」をクリックします。

ebXML レジストリへのプロデューサ詳細情報の発行

プロデューサを発行すると、Sun Java System Service Registry Server や ebXML Registry サーバーなどのリポジトリのいずれかにプロデューサの詳細情報が格納されます。プロデューサの発行後は、アプリケーションインタフェースまたはコマンド行インタフェースを使用してプロデューサの詳細情報を検索できます。Sun Java System Service Registry Server のセットアップに関する詳細については、『Service Registry 3.1 管理ガイド』を参照してください。

プロデューサの詳細情報をレジストリに発行するには、Sun Java System Portal Server for Registry を設定する必要があります。

▼ Sun Java System Portal Server for Registry を設定する

- 1 **Portal Server** がインストールされるマシンにディレクトリ `/soar/3.0/jaxr-ebxml/security` を作成します。
- 2 `keystore.jks` をレジストリサーバーの `/var/opt/SUNWsrvc-registry/3.0/data/security` ディレクトリから `/soar/3.0/jaxr-ebxml/security` にコピーします。
- 3 **Portal Server** 管理コンソールにログインします。
- 4 「ポータル」タブを選択します。
- 5 「ポータル」からポータルサーバーを選択します。
- 6 サブメニューの「SSOアダプタ」をクリックします。
- 7 「JES-REGISTRY-SERVER」をクリックします。
「メタアダプタの編集 - JES-REGISTRY-SERVER」画面が表示されます。
- 8 詳細情報を入力します。
プロキシを介してレジストリサーバーにアクセスしている場合

http.proxy.host	プロキシサーバーのホスト名。
http.proxy.password	プロキシサーバーが認証を必要とした場合のプロキシパスワード。
http.proxy.port	プロキシサーバーが有効となっているポート。
http.proxy.user	プロキシサーバーが認証を必要とした場合のプロキシユーザー名。

プロキシサーバーを使用していない場合

registry.keypassword	キーストアからキーを取得するために必要なパスワード。
registry.keystorealias	キーストアに存在するキーエイリアス。レジストリサーバーとの認証に使用されます。
registry.keystorelocation	キーストアの /soar/3.0/jaxr-ebxml/ からの相対的な場所。
registry.keystorepassword	キーストアを開くために使用するパスワード。
registry.publishurl	発行要求が送信されるレジストリサーバーの URL。この URL は SOAP 要求を受け入れるようにしてください。
registry.queryurl	検索要求が送信されるレジストリサーバーの URL。この URL は SOAP 要求を受け入れるようにしてください。

▼ プロデューサの詳細情報をレジストリに発行する

次の手順は、プロデューサをレジストリサーバーに発行する方法を示しています。

- 1 組織データファイルとプロデューサデータファイルを作成します。組織データファイルには次のエントリを含めることができます。

```
org.name= Sun Microsystems
org.description= Description
org.primarycontact.name= Henry
org.primarycontact.phoneno= 1234567
org.primarycontact.email= someone@host.com
```

注 - レジストリが内部に配備されていない限り、org.name および org.description は Identity Server の詳細情報と同じになるようにしてください。

プロデューサデータファイルには、次のエントリを含めるようにします。

```
producer.name=Producer_name
```

```
producer.description=Producer_Description
```

```
producer.id=Producer_ID
```

注-すべてのデータファイルを作成しなければいけないわけではありません。ただし、プロデューサ、組織、あるいはポートレットの詳細情報を検索するには、それに関連付けられたファイルを少なくとも1つ作成しておくようにしてください。

- 2 共通エージェントコンテナを停止し、再起動します。

```
/usr/lib/cacao/bin/cacaoadm stop
```

```
/usr/lib/cacao/bin/cacaoadm start
```

- 3 プロデューサの詳細情報を発行するには、次のコマンドを使用します。

```
./psadmin publish-registry -u amadmin -f password_file -p portal1 -m producer -U  
producer_data_file -O organization_data_file -T portlet - L --debug
```

注-ポートレットファイルは、WSRPプロデューサによって提供されるポートレットを指定します。ポートレットのリストは二重引用符で囲んだ文字列の形式で指定し、要素間をスペースで区切ります。たとえば、"NotepadPortlet BookmarkPortlet WeatherPortlet"のように指定します。

注-次のコマンドを使用すると、ログファイルをチェックできます。more
var/opt/SUNWportal/logs/admin/portal.admin.cli.0.0.log

参考 同等の psadmin コマンド

```
psadmin publish-registry
```

プロデューサの検索

ここではプロデューサの検索方法について説明します。

▼ プロデューサを検索する

- 1 検索プロデューサデータファイルを作成します。
検索プロデューサデータファイルには、次の内容を含めることができます。

producer.name= *producer_name*

producer.description= *producer_description*

注- 検索プロデューサデータファイルには、レジストリ内で検索対象となるプロデューサの説明が含まれます。文字列%をワイルドカードとして使用します。たとえば、%acme%を producer.name に使用すると、名前に文字列「acme」を含む WSRP プロデューサがすべて検索されます。

- 2 レジストリを検索するには、次のコマンドを使用します。

```
./psadmin search-registry -m consumer -u amadmin -f ps_password -C  
search_producer_datafile -p portal1
```

- 3 検索ポートレットデータファイルを作成します。

検索ポートレットデータファイルには、次の内容を含めることができます。

portlet.name= *portlet_name*

portlet.description= *portlet_description*

注- 検索ポートレットデータファイルには、レジストリ内で検索対象となるポートレットの説明が含まれます。文字列%をワイルドカードとして使用します。たとえば、%stock%を portlet.name に使用すると、名前に文字列「stock」を含むポートレットがすべて検索されます。

- 4 ポートレットの詳細情報を基に検索するには、次のコマンドを使用します。

```
./psadmin search-registry -m consumer -u amadmin -f ps_password -D  
search_portlet_datafile
```

- 5 検索組織データファイルを作成します。

検索組織データファイルには次の内容を含めてください。

organization.name= *organization_name*

organization.description= *organization_description*

注- 検索組織データファイルには、レジストリ内で検索対象となる組織の説明が含まれます。文字列%をワイルドカードとして使用します。たとえば、%acme%を organization.name に使用すると、名前に文字列「acme」を含む組織がすべて検索されます。

- 6 組織データファイルに基づいて検索するには、次のコマンドを使用します。

```
./psadmin search-registry -m consumer -u amadmin -f ps_password -L
search_organization_datafile -p portall
```

参考 同等の psadmin コマンド

```
psadmin search-registry
```

コンシューマの管理

ここでは、コンシューマ側で実行する必要がある操作について説明します。

この章で説明する内容は次のとおりです。

- 85 ページの「設定済みプロデューサの追加」
- 87 ページの「ID 伝播機構」
- 89 ページの「Web サービス SSO ポートレットを使用するユーザートークンプロファイルの作成」
- 88 ページの「ダイジェストパスワードの設定」
- 89 ページの「Web サービス SSO ポートレットを使用するユーザートークンプロファイルの作成」
- 89 ページの「サービスの説明の更新」
- 90 ページの「ユーザーカテゴリとロールのマッピング」
- 92 ページの「コンシューマ属性のマッピング」
- 92 ページの「プロキシの設定」

設定済みプロデューサの追加

プロデューサによって提供されたポートレットと通信するには、コンシューマは設定済みプロデューサを追加する必要があります。プロデューサが登録を必要とする場合は、次の方法を使用して設定済みプロデューサを追加します。

- 登録プロパティ値の入力 (in-band 登録)
- 登録ハンドルの入力 (out-of-band 登録)

プロデューサが登録を必要としない場合は、設定済みプロデューサの追加時にコンシューマは詳細情報を入力する必要はありません。

▼ 設定済みプロデューサを追加する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」タブをクリックします。
- 5 **DN**を選択し、「新規」をクリックします。
- 6 設定済みプロデューサの名前を入力します。**ID 伝播機構**を選択します。デフォルトでは「なし」が選択されています。

注-ID 伝播機構によって、コンシューマポータルのユーザーは自らの資格をプロデューサポータルに示すことができます。これは、ユーザーがIDをコンシューマポータルからプロデューサポータルに連携させることができる機構です。

- 7 **WSDL URL**を入力し、「次へ」をクリックします。

注-プロデューサまたはポートレットに基づいてWSDL URLを検索することもできます。プロデューサが発行されている場合に限り、検索結果にプロデューサのWSDL URLが表示されます。

- 8 プロデューサが登録を必要とする場合は、2つの方法でプロデューサを登録できます。すなわち、登録プロパティ値を入力する方法 (**in-band** 登録)と登録ハンドルを入力する方法 (**out-of-band** 登録)です。「次へ」をクリックします。
- 9 手順7で1番目の方法を選択した場合は、登録プロパティ値を入力し、「次へ」をクリックします。2番目の方法を選択した場合は、**out-of-band** 通信を介して取得した登録ハンドルを入力し、「次へ」をクリックします。
- 10 詳細情報を確認し、「完了」をクリックします。

参考 同等の psadmin コマンド

```
psadmin create-configured-producer
```

ID 伝播機構

ID 伝播は、WSRP コンシューマがユーザーの ID を WSRP プロデューサ Web サービスに提供するために使用する機構です。これは、ユーザーがコンシューマとプロデューサの間で ID を連携させるための連携機構です。連携に成功すると、コンシューマポータルはユーザー ID をプロデューサポータルに伝播します。WSRP プロデューサは、コンシューマからユーザー資格を受け取ると、資格を検証し、指定されたユーザーコンテキスト内のリソースへのアクセスを許可または拒否します。

ユーザーはポータルごとに 2 つの ID を持ちます。すなわち、1 つはプロデューサポータル用で、もう 1 つはコンシューマポータル用です。ユーザーは、提供された ID 伝播機構を使用して ID を連携させます。これによって、コンシューマポータルとプロデューサポータルに対する機構にシングルサインオンが提供されます。ユーザーがコンシューマポータルを介してポータルにログインするときには、ユーザーはプロデューサポータルに直接ログインするときに取得する内容を取得します。ユーザーがプロデューサポータルにログインするときには、ユーザーは連携された ID を使用して変更を行うことができます。

Sun Java System WSRP プロデューサは、次の ID 伝播をサポートします。

- SSO トークン: プロデューサポータルとコンシューマポータルの両方が同じ Access Manager インスタンスに接続されている場合に選択します。通常、プロデューサポータルとコンシューマポータルの両方が同じ組織内に配備されている構成で推奨されます。
- WSS UsernameToken プロファイル (ユーザー名のみ): ユーザー名が WS Security ヘッダーとしてコンシューマポータルからプロデューサポータルに伝播される WSS 仕様を使用します。
- WSS UsernameToken プロファイル (PasswordDigest 付き): WS Security ヘッダーが、プロデューサにおけるターゲットとされるユーザー ID をパスワードと共に Digest 形式で送信します。
- WSS UsernameToken プロファイル (PasswordText 付き): WS Security ヘッダーが、プロデューサにおけるターゲットとされるユーザー ID をパスワードと共に Text 形式で送信します。

上記のうち下の 3 つのオプションでは、OASIS WSS UsernameToken プロファイル仕様が実装されます。この仕様は、UsernameToken を Web サービスで使用方法を記述しています。WSS 仕様は、ユーザー名によって要求者を識別することによって、またオプションでパスワードを使用して ID を Web サービスプロデューサに対して認証することによって、Web サービスコンシューマがユーザー名トークンをどのように提供できるかを記述しています。

注 - 多くのポータルベンダーは、OASIS WSS UsernameToken プロファイル仕様をサポートおよび実装しています。相互運用性が必要とされる場合には、これら3つのオプションのいずれかを使用します。

Portal Server では ID 伝播機構に2つのレベルが存在します。まず、コンシューマポータルの管理者は、プロデューサポータルが上記で指定された ID 伝播機構のいずれかをサポートすることを発見します。管理者は、ユーザーに ID の送信を許可することができます。Portal Server コンシューマは、前述した ID 伝播機構のすべてをサポートします。

コンシューマの作成後、管理者はコンシューマがサポートする ID 伝播機構に基づいてリモートチャネルを作成する必要があります。チャネルがユーザーデスクトップで使用可能になると、ID 伝播を受け入れる準備が整います。

ID 伝播機構はプロデューサで自動的に設定されます。Portal Server は、Sun SSO、OASIS UsernameToken プロファイル、ID 伝播なしのモードの順に認証をチェックします。

ダイジェストパスワードの設定

新しいユーザーのみが、`configuration` コマンドを実行して LDAP パスワードをプレーンテキストに格納したあとにダイジェストパスワード機能を使用できます。

コンシューマの作成には、ユーザー ID 伝播機構に対して WSSO UsernameToken プロファイル (PasswordDigest 付き) オプションの選択を含めるようにしてください。

Web サービス SSO ポートレットを編集して、適切な Web サービス URL (プロデューサ) を選択し、新しいユーザー名とパスワードを提供する必要があります。

▼ ダイジェストパスワードの受け入れを設定する

ダイジェストパスワードを受け入れるように Sun Java System WSRP プロデューサを設定するには、次の手順に従います。

- 1 コマンド `/opt/SUNWdsee/ds6/bin/dscfg set-server-prop pwd-storage-scheme: CLEAR` を実行して、プレーンテキストパスワードが格納されるように、**Directory Server** のパスワード格納方式を変更します。

注 - Directory Server のデフォルトのインストール場所が `/opt/SUNWdsee` であることを想定しています。

- 2 **UsernameToken** プロファイル (**PasswordDigest** 付き) を使用できるように、**AM** コンソールで新しいユーザーを作成します。

参考 推奨事項

- WSS **UsernameToken** プロファイル (**PasswordDigest** 付き) を使用する場合、パスワードがプレーンテキストで送信されるため、プロデューサポータルとコンシューマポータル間の通信をセキュリティー保護するようにしてください。
- 同じプロデューサ URL を指す2つの異なるコンシューマは、同じ種類の ID 伝播機構を使用するようにしてください。

Web サービス **SSO** ポートレットを使用するユーザートークンプロファイルの作成

ユーザーが ID 伝播機構を使用する場合にユーザーの資格を認証するためのユーザートークンプロファイルを作成することができます。プロデューサが提供する特定の Web サービスに対するユーザー名とパスワードを定義できます。

▼ **Web** サービス **SSO** ポートレットを使用するユーザー資格を提供する

- 1 **Portal Server** デスクトップにログインします。
- 2 「**Web** サービス **SSO** ポートレット」で、「**編集**」ボタンをクリックします。
- 3 「新しいユーザートークンプロファイルを作成」セクションで、ユーザートークンプロファイルを作成する「**Web** サービスの URL」を選択します。
- 4 ユーザー名とパスワードを入力します。「**追加**」をクリックします。
既存のユーザートークンプロファイルを編集および削除することも可能です。

サービスの説明の更新

コンシューマによるプロデューサの設定後には、その後のプロデューサに対する変更をすべて更新するように、「サービスの説明の更新」オプションを使用します。たとえば、新しいポートレットの追加や、登録後の登録プロパティーの変更などです。

▼ サービスの説明を更新する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」タブをクリックします。
- 5 **DN** (識別名) を選択します。
- 6 設定済みプロデューサのリンクをクリックします。
- 7 「設定済みプロデューサを編集」画面で、「サービスの説明の更新」をクリックします。

参考 同等の psadmin コマンド

```
psadmin update-configured-producer-service-description
```

ユーザーカテゴリとロールのマッピング

WSRP はユーザーカテゴリのコンセプトをサポートしています。これは、プロデューサのサービスの説明に含まれます。ユーザーカテゴリをロールにマッピングすると、ユーザーはコンシューマポータルで定義されているロールをポートレットで定義されているロールにマッピングすることができます。Sun Java System Portal Server は Java System Access Manager のロールをポートレットのロールにマッピングします。これらのロールを、対応する WSRP ユーザーカテゴリにマッピングすることができます。

次の作業を行うことができます。

- [91 ページの「ポートレットでロールを作成する」](#)
- [92 ページの「ユーザーカテゴリをロールにマッピングする」](#)

ロールの定義は、ポートレットの配備中に行うことができます。

注- ポートレットで定義されたロールは、プロデューサの Access Manger 内に存在しなければいけません。

▼ ポートレットでロールを作成する

次の作業では、Sun Java System Access Manager の amconsole とポートレットにロールが作成されます。

- 1 Access Manager コンソールにログインします。
- 2 ロールを作成し、そのロールにユーザーを追加します。
- 3 ポートレットアプリケーションの webxml に、次のコードを追加します。

```
<security-role>  
  
<role-name>PS_TEST_DEVELOPER_ROLE</role-name>  
  
</security-role>
```

- 4 ポータルの portlet.xml に次の行を追加します。

```
<security-role-ref>  
  
<role-name>PS_TEST_DEVELOPER_ROLE</role-name>  
<role-link>PS_TEST_DEVELOPER_ROLE</role-link>  
  
</security-role-ref>
```

- 5 ポートレットアプリケーションの war ファイルを作成します。

- 6 次のエントリを含むロールファイルを作成します。

```
cn\=AM_TEST_DEVELOPER_ROLE,o\=DeveloperSample,dc\=india,dc\=sun,dc\  
=com=PS_TEST_DEVELOPER_ROLE
```

- 7 次のコマンドを使用して、ポートレットを配備します。

```
/opt/SUNWportal/bin/psadmin deploy-portlet -u amadmin -f ps_password -d  
"o=DeveloperSample,dc=india,dc=sun,dc=com" -p portal1 -i stockprice-8080 --rolesfile  
rolesfile TestPortlet.war
```

参考 同等の psadmin コマンド

```
psadmin deploy-portlet
```

▼ ユーザーカテゴリをロールにマッピングする

ユーザーカテゴリをロールにマッピングするには、次の手順に従います。

- 1 「コンシューマ」タブでプロデューサ名のリンクをクリックします。
「設定済みプロデューサを編集」画面に次の内容が表示されます。「ユーザーカテゴリ」:プロデューサポートレット内のロール。「ローカルロール」:コンシューマの Sun Java System Access Manager で定義されているロール。
- 2 「ユーザーカテゴリとロールのマッピング」セクションで、コンシューマで定義されているロールにユーザーカテゴリをマッピングし、「了解」をクリックします。

コンシューマ属性のマッピング

Sun Java System Portal Server の WSRP コンシューマの実装は、Sun Java System Directory Server 上のユーザーエントリに格納されている共通ユーザー属性を、WSRP 仕様が要求するユーザー属性の標準セットにマッピングします。

コンシューマポートレットがLDAPスキーマで指定されていない属性を使用する場合は、それらの属性を格納するカスタムオブジェクトクラスを作成し、このオブジェクトクラスをユーザーエントリに追加します。属性の作成後に、Sun Java System Access Manager 管理コンソールを使用して、LDAP 属性を対応する WSRP 属性にマッピングします。

プロキシの設定

コンシューマおよび Web コンテナ XML ファイルには、プロキシを設定する必要があります。

次の作業を行うことができます。

- [92 ページの「共通エージェントコンテナ内のコンシューマのプロキシを設定する」](#)
- [93 ページの「Web コンテナ XML ファイルを設定する」](#)

▼ 共通エージェントコンテナ内のコンシューマのプロキシを設定する

- 1 `./cacaoadm get-param java-flags` を実行します。
- 2 値をコピーし、`./cacaoadm set-param java-flags` にペーストします。

- 3 次の内容をコマンドに追加します。-Dhttp.proxyHost= *webcache.canada.sun.com*
-Dhttp.proxyPort=8080 -Dhttp.proxyUser= *Proxyuser* -Dhttp.proxyPassword=*Password*
- 4 **Enter** キーを押します。
- 5 共通エージェントコンテナサーバーを再起動します。

▼ Web コンテナ XML ファイルを設定する

- 1 次のファイルを編集します。

```
vi /var/opt/SUNWappserver/domains/domain1/config/domain.xml
```
- 2 次の JVM オプションを設定します。
 - Dhttp.proxyHost
 - Dhttp.proxyPort
 - Dhttp.proxyUser
 - Dhttp.proxyPassword

WSRP プロデューサの管理

ここでは、Sun Java System Portal Server WSRP (Web Services for Remote Portlets) サービスの管理方法について説明します。WSRP プロデューサの管理作業は次のとおりです。

- [93 ページの「WSRP プロデューサを作成する」](#)
- [94 ページの「WSRP プロデューサを編集する」](#)
- [95 ページの「コンシューマ登録を作成する」](#)
- [96 ページの「コンシューマ登録を編集する」](#)

▼ WSRP プロデューサを作成する

WSRP プロデューサの作成には、次の内容を使用します。

- プロデューサインスタンスの名前 (ポータルサーバー全体で一意でなければならない)
- 登録が必要かどうか。「登録が必要」を選択した場合、すべての WSRP コンシューマは要求を出す前に、このプロデューサインスタンスに登録する必要があります。登録されていない WSRP コンシューマからの要求は拒否されます。

- in-band 登録がサポートされているかどうか。in-band 登録によって、WSRP コンシューマのプログラムでの登録が可能になります。一方、out-of-band 登録では、登録ハンドルへのアクセス権の設定および交換のために、WSRP コンシューマ管理者と WSRP プロデューサ管理者の間で手動による連絡(電子メール、電話など)が必要です。

- 1 **Portal Server 管理コンソールにログインします。**
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「プロデューサ」をクリックします。
- 5 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 6 「WSRP プロデューサ」から「新規」をクリックしてウィザードを起動します。
- 7 指示に従って、指定したプロデューサを作成します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

参考 同等の psadmin コマンド

```
psadmin create-producer
```

▼ WSRP プロデューサを編集する

次の手順に従って、WSRP プロデューサを編集できます。

- 発行済みリストからポートレットを追加または削除します。
- 登録の要件を変更します。



注意-このオプションの変更は既存のプロデューサに対して行ってください。

- in-band 登録を有効または無効にします。
- 登録検証クラスを指定します。登録検証クラスは、WSRP プロデューサが WSRP コンシューマが送信した値を受諾できるかどうかを検証するために使用します。
- 新しい登録プロパティを追加します。プロパティの変更は、その後のプロデューサへのコンシューマの登録にすべて適用されます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「プロデューサ」をクリックします。
- 5 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
- 6 **WSRP** プロデューサを選択し、必要に応じて設定属性を変更します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。
- 7 「保存」をクリックして、変更を記録します。

参考 同等の psadmin コマンド

```
psadmin set-attribute
```

▼ コンシューマ登録を作成する

各コンシューマ登録は、WSRP プロデューサとの関係を確立したりリモートの WSRP コンシューマを表します。サポートする WSRP プロデューサには、複数の WSRP コンシューマを登録できます。この登録機構によって、WSRP コンシューマは自身の機能を WSRP プロデューサに説明できます。

WSRP コンシューマは out-of-band で追加されます (電子メールや電話などによって)。コンシューマ登録を追加するときに入力される情報は、登録ハンドルを与えられる WSRP コンシューマの機能と一致する必要があります。コンシューマ登録を行うと、WSRP プロデューサは、WSRP コンシューマが WSRP プロデューサ上に作成する成果物 (ポートレット設定など) をスコープできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「プロデューサ」をクリックします。
- 5 「DNを選択」ドロップダウンメニューから任意のDNを選択します。

- 6 WSRP プロデューサを選択し、「コンシューマ登録」を選択します。
- 7 「新規」をクリックしてウィザードを起動します。
- 8 指示に従って、指定したコンシューマ登録を作成します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

参考 同等の psadmin コマンド

```
psadmin create-consumer-registration
```

▼ コンシューマ登録を編集する

既存のコンシューマ登録を手動で編集できます。これは、WSRP コンシューマ側から in-band 登録によって行うことも可能です。out-of-band と in-band の登録を同時に使用することはできません。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「プロデューサ」をクリックします。
- 5 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 6 プロデューサ、WSRP プロデューサ、「コンシューマ登録」の順に選択します。
- 7 コンシューマ登録を選択し、必要に応じて設定属性を変更します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。
- 8 「保存」をクリックして、変更を記録します。

WSRP コンシューマの管理

ここでは、WSRP コンシューマを管理する作業について説明します。

- 97 ページの「設定済みプロデューサを追加する」
- 97 ページの「設定済みプロデューサを編集する」
- 98 ページの「コンシューマ名を指定する」

▼ 設定済みプロデューサを追加する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「プロデューサ」をクリックします。
- 5 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 6 「設定済みプロデューサ」の下の「新規」をクリックしてウィザードを起動します。
- 7 指示に従って、指定した設定済みプロデューサを作成します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

参考 同等の psadmin コマンド

```
psadmin create-configured-producer
```

▼ 設定済みプロデューサを編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「WSRP」をクリックし、サブメニューから「コンシューマ」をクリックします。

- 5 「DN を選択」 ドロップダウンメニューから任意の DN を選択します。
- 6 設定済みプロデューサを選択し、必要に応じて設定属性を変更します。

注- 「サービスの説明の更新」 オプションを使用して、プロデューサに対する変更をすべて更新します。89 ページの「サービスの説明の更新」を参照してください。

属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

- 7 「保存」 クリックして、変更を記録します。

参考 同等の psadmin コマンド

```
psadmin set-attribute
```

▼ コンシューマ名を指定する

WSRP コンシューマは、登録作業中にコンシューマ名をプロデューサに送信します。ここでコンシューマ名として指定された値は、組織またはサブ組織レベルでほかの値がコンシューマ名として指定されないかぎり、デフォルトとして使用されません。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」 タブを選択します。
- 3 「ポータル」 からポータルサーバーを選択します。
- 4 「WSRP」 をクリックし、サブメニューから「コンシューマ」 をクリックします。
- 5 「DN を選択」 ドロップダウンメニューから任意の DN を選択します。
- 6 「WSRP コンシューマ」 の下の「編集」 をクリックします。
- 7 コンシューマ名を指定します。
- 8 「了解」 をクリックします。

参考 同等の psadmin コマンド

```
psadmin set-attribute
```

Portal Server エンドユーザー動作の追跡の管理

この章では、Sun Java System Portal Server ユーザー動作の追跡方法について説明します。

この章で説明する内容は次のとおりです。

- 99 ページの「Portal Server ユーザー動作の追跡の概要」
- 101 ページの「Portal Server ユーザー動作の追跡のセットアップ」

Portal Server ユーザー動作の追跡の概要

Portal Server ユーザー動作の追跡 (UBT) により、Portal Server アプリケーション上のエンドユーザー動作を追跡する手段が提供されます。ポータルデスクトップ上のユーザー操作は ubt ログファイルに収集されます。ubt ログファイルは、W3C 標準の拡張ログファイル形式で記録されます。このログファイルを元に、Portal Server コンソールまたは `psadmin generate-ubt-report` コマンドを使用して、さまざまなエンドユーザー動作の追跡レポートを作成できます。また、AWStats などのサードパーティーのツールを使用して、UBT レポートを生成することもできます。

UBTConfig.properties ファイルでも UBT を有効にすることができます。
`/var/opt/SUNWportal/portals/portalID/config/UBTConfig.properties` で
`com.sun.portal.ubt.enable=true` を設定します。

次の表は、UBT レポート、その説明、および有効なレポート形式の一覧です。

表 6-1 ユーザー動作の追跡レポート

レポート名	レポートの説明	レポートの形式
Portal ユーザー ID レポート	このレポートには、各ユーザーの最終ポータルアクセス時刻が表示されます。ユーザーは、アクセスしたサーバー、所属するドメイン、および相対 DN ごとにグループ化されます。	HTML または PDF
Portal ユーザーログイン率	このレポートには、ポータルへのログイン率が表示されます。	
Portal チャネル表示レポート	このレポートには、チャネルを表示中の各ユーザーが当該チャネルを表示した回数が表示されます。チャネルは、所属するコンテナごとにグループ化されます。	HTML または PDF
Portal コンテナのユーザーカスタマイズ	このレポートには、Portal コンテナのカスタマイズが表示されます。通常、コンテナのカスタマイズでは、デスクトップ上の「コンテンツ」、「レイアウト」、または「テーマ」の変更点を参照します。	HTML または PDF
Portal 要求率	このレポートには、一定期間における各最上位コンテナの1時間ごとの要求率が表示されません。最上位コンテナの要求をページ要求と見なすことができます。	HTML または PDF
Portal チャネルのユーザーカスタマイズ	このレポートには、各エンドユーザーがチャネル上で実行したアクションが表示されます。ユーザーは、アクセスしたコンテナ、およびアクションを実行したチャネルごとにグループ化されます。	HTML または PDF
Portlet アクションレポート	このレポートには、Portal でのポートレットアクション要求率が表示されます。	HTML または PDF

表 6-1 ユーザー動作の追跡レポート (続き)

レポート名	レポートの説明	レポートの形式
ポートレットレンダリングレポート	このレポートには、特定のウィンドウ状態中にポートレットがポートレットモードで表示される回数が表示されます。ウィンドウ「最小化」状態では、ポートレットはレンダリングされません。したがって、この状態での回数は表示されません。	HTML または PDF

Portal Server ユーザー動作の追跡のセットアップ

ここでは、ユーザー動作の追跡を有効にする方法とレポートの生成方法について説明します。

Portal Server 管理コンソールで次の作業を行うことができます。

- 101 ページの「ユーザー動作の追跡ログを有効にする」
- 102 ページの「ユーザー動作の追跡レポートを生成する」

▼ ユーザー動作の追跡ログを有効にする

デフォルトでは、Portal Server アプリケーション上の UBT ログは有効になっていません。

- 1 Portal Server 管理コンソールにログインします。
- 2 「共通タスク」タブを選択します。
- 3 「レポートおよびログ」の下に「ポータル使用状況レポート」をクリックしてウィザードを起動します。
- 4 「ポータルを選択」ドロップダウンメニューでポータルインスタンスを選択し、「了解」をクリックします。
「ユーザー動作の追跡」ページが表示されます。
- 5 「設定」サブメニューをクリックし、「共通プロパティ」の下に「UBT ログ」を有効にします。

共通プロパティ、ハンドラプロパティ、およびイベント設定の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

注-その他すべてのプロパティーについては、UBTを動作させるのに適切なデフォルト値があらかじめ設定されています。Portal Server のすべてのインスタンスに対する変更を適用するには、「すべてのインスタンスに適用」ボタンをクリックします。一部のインスタンスの場合は、「選択したインスタンスに適用」ボタンをクリックします。

- 6 ポータルデスクトップにアクセスし、ユーザー動作の追跡ログファイルが生成されていることを確認します。
デフォルトでは、ユーザー動作の追跡ログは
`/PortalData-Dir/portals/PortalID/logs/instanceID/ubt.0.0.log` ファイルに書き込まれます。

▼ ユーザー動作の追跡レポートを生成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「共通タスク」タブを選択します。
- 3 「レポートおよびログ」の下の「ポータル使用状況レポート」をクリックしてウィザードを起動します。
- 4 「ポータルを選択」ドロップダウンメニューでポータルインスタンスを選択し、「了解」をクリックします。
「ユーザー動作の追跡」ページが表示されます。
- 5 「レポート」サブメニューをクリックします。
8種類のレポートが一覧表示されます。これらすべてのレポートをPDFまたはHTMLの形式で生成できます。詳細については、[表 6-1](#) を参照してください。

参考 同等の psadmin コマンド

```
psadmin generate-ubt-report
```

Portal Server の動作の監視

この章では、Sun Java™ System Portal Server 監視の設定方法について説明します。

この章で説明する内容は次のとおりです。

- 103 ページの「Portal Server 監視について」
- 104 ページの「Portal Server 監視の設定」
- 105 ページの「Portal Server 監視データの収集」

Portal Server 監視について

監視によって、ポータルサーバーに関する実行時リソース情報を記録できます。デスクトップ監視は、ポータルサーバーから受信した要求に基づいてコンテンツ、編集、プロセスタイプの情報を記録し続けます。また、ポータルサーバーのさまざまなチャンネルの各タイプの要求に対して、最短、最長、平均応答時間についての情報を記録します。

ポータルの動作の監視によって収集された情報は、より長い応答時間が必要なチャンネルを別のセカンダリタブに移動するか、キャッシュヒットに基づいてデスクトップチャンネルのタイムアウトプロパティを設定することによって、ポータルの応答時間を最適化するために役立ちます。

ポータルサーバーの Java 仮想マシン (JVM) は、デスクトップの監視データを収集します。監視情報は、Portal Server 管理コンソールで表示するか、psadmin 監視サブコマンドを使用してアクセスできます。『Sun Java System Portal Server 7.1 Command Line Reference』を参照してください。

監視には、Java Management Extensions (JMX™ テクノロジ) を使用し、ポータルサーバーデスクトップとポータルデスクトップチャンネルを示すポータルサーバーインスタンスの MBeansServer で Management Beans (MBeans) を登録します。各 MBean 属性は、各リソースに対して収集した監視データを示します。ポータル管理コンソールと psadmin 監視サブコマンドは、MBeans と通信し、ポータルサーバーインスタンスの監視データを収集し、表示します。

Portal Server 監視の設定

監視は、

`/var/opt/SUNWportal/portals/portalID/config/instanceID/monitoring.properties` ファイルに保存された監視プロパティにアクセスして設定できます。監視はデフォルトで有効になっています。監視を無効にするには、`com.sun.portal.monitoring.MonitoringContext.monitoring.disable` プロパティを「true」に設定します。JVMが再起動すると、監視は無効になります。

ポータル管理コンソールで監視を有効または無効にすることもできます。

- 104 ページの「ポータル監視を有効または無効にする」
- 104 ページの「デスクトップ統計を表示する」
- 105 ページの「チャンネル統計を表示する」

▼ ポータル監視を有効または無効にする

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「監視」タブをクリックします。
- 5 「設定」サブメニューをクリックします。
- 6 ポータルサーバーインスタンスを選択します。
- 7 「監視を有効」ボタンまたは「監視を無効」ボタンをクリックします。

▼ デスクトップ統計を表示する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「監視」タブをクリックします。
- 5 サブメニューから「デスクトップ要求/応答統計」をクリックします。

▼ チャネル統計を表示する

- 1 Portal Server 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」でポータルサーバーを選択します。
- 4 「監視」タブをクリックします。
- 5 サブメニューの「チャネルアクション統計」をクリックします。
- 6 「DNを選択」ドロップダウンメニューで組織を選択します。
- 7 「サーバーインスタンス」ドロップダウンメニューからサーバーを選択します。

Portal Server 監視データの収集

監視によって、デスクトップから受信する7種類のデータ要求を収集できます。どのタイプの要求も、要求タイプとしてタイプ `DesktopRequestStatistic` と名前 MBean プロパティを持つ MBean として表示されます。たとえば、`type=DesktopRequestStatistics,name=Content` 名前プロパティは、デスクトップコンテンツ要求統計を特定するために役立ちます。

デスクトップ統計

次のリストは7種類の要求を説明しています。

Content	デスクトップがコンテンツ要求を正常に処理した回数と、それにかかった時間。
Edit	デスクトップが編集要求を正常に処理した回数と、それにかかった時間。
Exception	要求処理中のいくつかの例外によって、デスクトップが要求を処理できなかった回数。例外情報は、ポータルサーバーログファイルに記録されます。
LocalAuth	デスクトップがローカル認証要求に応答した回数。
Logout	ユーザーがポータルサーバーからログアウトした回数と、ログアウトにかかった時間。
PreLogin	デスクトップが PreLogin 要求に応答した回数。

Process デスクトップが編集要求を処理した回数と、それにかかった時間。

チャンネル統計は、ポータル管理コンソールから表示できます。

チャンネル統計

各タイプのチャンネル動作は、チャンネルを特定する追加の名前プロパティーとともに、タイプ `ChannelActionStatistic` の MBean として示されます。完全な MBean 名を知るには、コマンド `command psadmin get-monitoring-mbean-names` を使用します。

ポータルデスクトップは、タイムアウトチャンネルプロパティーに基づいてキャッシュされたチャンネルのコンテンツ表示を表示します。

次のリストは、各デスクトップチャンネルで監視されるチャンネル動作のタイプを説明しています。

Content チャンネルプロバイダがコンテンツ表示を正常に生成した回数とその応答時間。

Edit チャンネルプロバイダが編集表示を正常に行った回数とその応答時間。

Process チャンネルプロバイダが編集表示を処理した回数。

チャンネル統計は、ポータル管理コンソールから表示できます。

Portal Server のロギングの管理

この章では、Sun Java™ System Portal Server ログ情報の取得方法について説明します。

この章で説明する内容は次のとおりです。

- 107 ページの「Portal Server のロギングについて」
- 107 ページの「Portal Server のロギングの管理」

Portal Server のロギングについて

Portal Server は、すべてのコンポーネントのロギングをサポートします。ログとログ設定はすべてのポータルコンポーネントで同一です。簡単なものから詳細なものまで7つの標準のログレベルがあります。ログは、異なるファイルや受信端末装置に配信でき、1つのファイル、または各コンポーネントに1つずつの複数のファイルで構成できます。

ログレベルは、モジュールとサブモジュールごとに設定でき、ログは各コンポーネント内の各モジュールとサブモジュールに対して個別のファイルに配信できます。

Portal Server のロギングの管理

次のコンポーネントを使用して、Portal Server のロギングを設定および管理できます。

- ログビューア
- 共通ロガーの設定
- 個別ロガーの設定

ポータルのロギングはポータル管理コンソールから管理できます。

- 108 ページの「ログビューアを管理する」
- 109 ページの「ログ表示をカスタマイズする」

- 109 ページの「共通ロガーの設定を管理する」
- 111 ページの「個別ロガーの設定を管理する」

▼ ログビューアを管理する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」の下のポータルサーバーを選択します。
- 4 「ロギング」をクリックして、サブメニューから「ログビューア」をクリックします。
- 5 「インスタンス名」ドロップダウンメニューからポータルインスタンスを選択します。

ログビューアの「検索条件」および「検索結果」ページが表示されます。

- 6 「検索条件」の値を入力して、「検索」をクリックします。

次の検索オプションが使用できます。

ログファイル名 ログコンテンツを含むファイル名。

ログレベル 選択したレベルかそれ以上のレベルのメッセージがログに表示されます。使用可能なレベルは、SEVERE、警告、情報、設定、FINE、FINER、およびFINESTです。デフォルトレベルは、情報です。このため、ログには、情報、警告、またはSEVEREレベルのメッセージが含まれます。

表示したいメッセージがログに必ず表示されるように、「個別ロガーの設定」ページで最初に適切なログレベルを設定します。

タイムスタンプ 特定の期間のログメッセージを表示します。

100 の最新のログエントリが表示できます。または「開始日」および「終了日」テキストボックスに期間を入力します。

「範囲」を選択した場合、

- 「開始日」と「終了日」の両方の値が必要です。
- 「開始日」の値は「終了日」より遅い日付にできません。
- 「終了日」の値は今日より遅い日付にできません。

- 「開始日時刻」と「終了日時刻」の値はオプションです。「開始日時刻」が指定されている場合、「終了日時刻」を指定する必要があります。時刻の値には、構文を次の形式にする必要があります。hh:mm:ss.SSS。SSS はミリ秒を示します。たとえば、18:20:10.000 のようになります。

参考 同等の psadmin コマンド

```
psadmin set-logger
```

▼ ログ表示をカスタマイズする

次の手順で「検索結果」ページをカスタマイズできます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」の下の **Portal Server** を選択します。
- 4 「ロギング」をクリックしてから、「インスタンス名」ドロップダウンメニューでポータルサーバーを選択します。
- 5 「ログビューアの結果」テーブルで「タイムスタンプ」列ヘッダをクリックし、メッセージをソートします。
- 6 詳細リンクをクリックして、新しいウィンドウに書式化したログメッセージを表示します。

▼ 共通ロガーの設定を管理する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」の下の **Portal Server** を選択します。
- 4 「ロギング」をクリックし、サブメニューから「共通ロガーの設定」をクリックします。

- 5 「インスタンス名」ドロップダウンメニューからポータルインスタンスを選択します。
- 6 必要に応じて設定属性を変更します。
次のオプションが使用できます。

一般

ログレベル — ログレベル設定を選択して、ログファイルに表示する情報を選択できます。

ログレベルの選択肢は次のとおりです。

- Severe - ユーザーに表示されるエラー
- 警告 - ユーザー警告
- 情報 - ユーザーに対する情報
- 設定 - 開発者に対する静的な設定情報
- Fine - 基本的なトレース情報
- Finer - 詳細なトレース情報
- Finest - 完全なトレース情報
- Off - ロギングをオフにする場合に使用できる
- すべて - すべてのメッセージをログ記録することを意味します

ファイルハンドラのプロパティ

- 制限 — ログファイルのサイズをバイト数で指定します。ログファイルのサイズがこの値を超えると、ログファイルはファイル数に応じてローテーションされます。デフォルト値は5Mバイトです。
- ファイル数 — ログファイルがバイトで指定されたサイズに達すると、世代番号(ファイルパターンの %g) が1つ増えた新しい空のファイルが作成されます。デフォルト値は2です。ログファイルのローテーションをオフにするには、値を0に設定します。
- 追加 — 新しいメッセージを既存のファイルに追加するかどうかを指定します。デフォルトは、「true」です。
- フィルタ — ポータルログのような宛先やカスタムログハンドラによって指定された宛先に送られるログ記録をフィルタするために、カスタムログフィルタプラグインを使用できます。カスタムフィルタは、インタフェース `java.util.logging.Filter` を実装

する必要があります。フィールドにフィルタの絶対クラス名を入力します。また、アプリケーションサーバークラスパスにフィルタクラスを含めて、フィルタがサーバーの起動中にインストールされるようにします。

その他

- カスタムハンドラー ポータルログ以外の宛先にログを送るために、カスタムログハンドラーに接続できます。カスタムハンドラーはクラス `java.util.logging.Handler` を拡張する必要があります (JSR 047 準拠 API)。フィールドにハンドラーの絶対クラス名を入力します。また、アプリケーションサーバークラスパスにハンドラークラスを含めて、ハンドラーがサーバーの起動中にインストールされるようにします。複数のハンドラーを指定できます。複数の名前を区切るには、コンマを使用します。
 - Web コンテナログファイルを使用 — ポータルロギング管理を無効にして、すべてのログを Web コンテナログファイルに配信するには、「はい」を選択し、それ以外の場合には「いいえ」を選択します。デフォルトは、「いいえ」です。
- 7 「選択したインスタンスに適用」または「すべてのインスタンスに適用」をクリックして、変更を記録します。

参考 同等の psadmin コマンド

```
psadmin set-logger
```

▼ 個別ロガーの設定を管理する

- 1 [Portal Server 管理コンソールにログインします。](#)
- 2 「ポータル」タブを選択します。
- 3 「ポータル」から **Portal Server** を選択します。

- 4 「ロギング」をクリックし、サブメニューから「個別ロガーの設定」をクリックします。
- 5 「インスタンス名」ドロップダウンメニューからポータルインスタンスを選択します。
- 6 必要に応じて設定属性を変更します。
次のオプションが使用できます。

ロガー設定

- ロガー名 - ロガー名をクリックして、ロガーの設定詳細を表示します。
- ログレベル - ログレベル設定を選択してロガーのログファイルに表示する情報を選択するか、親ロガーからログレベルを継承できます。たとえば、`debug.com.sun.portal` のログレベルが「情報」で、`debug.com.sun.portal.desktop` のログレベルが「親ロガーのレベルを継承」である場合、この値も「情報」になります。
- ログファイルのマージ方針 - ロガーに対して、親と同じログファイルにメッセージをログ記録するか（「親ログファイルにログを記録」）、ログを別のファイルに記録するか（「個別ログファイルにログを記録」）を選択できます。
- 親ハンドラ - ロガーに対して、「ログファイルのマージ方針」が「個別ログファイルにログを記録」に設定されている場合、個別ログファイルと親ログファイルの両方にメッセージをログ記録するか（「親ハンドラを継承する」）、個別のファイルのみにログ記録するか（「親ハンドラを継承しない」）を選択できます。
- スタックトレース - ロガーに対して、スタックトレースをすべてのレベルで記録するか（「すべてのレベルのスタックトレースを出力」）、警告ログレベルまでのみを記録するか（「警告レベルまでスタックトレースを出力」）を選択できます。

注 - 「ログファイルのマージ方針」の値が「親ログファイルにログを記録」の場合、「親ハンドラ」と「スタックトレース」の値は無視されます。「ログファイルのマージ方針」の値が「個別ログファイルにログを記録」で、「親ハンドラ」の値が「親ハンドラを継承する」の場合、「すべてのレベルのスタックトレースを出力」という「スタックトレース」の値は無効です。

- 7 「選択したインスタンスに適用」または「すべてのインスタンスに適用」をクリックして、変更を記録します。

参考 同等の psadmin コマンド

```
psadmin set-logger
```


Portal Server 登録の管理

この章では、Sun Java™ System Portal Server 登録コンポーネントとその管理方法について説明します。この章で説明する内容は次のとおりです。

- 115 ページの「Portal Server 登録の概要」
- 116 ページの「登録のセットアップ」
- 121 ページの「Portal Server ディスカッションの管理」

Portal Server 登録の概要

登録を使用することで、エンドユーザーはカテゴリ、ディスカッション、検索可能なドキュメントなど多くの情報ソースを網羅するプロファイルを作成できます。エンドユーザーが登録チャンネルにアクセスするたびに、プロファイルは最新の情報で更新されます。登録チャンネルは、エンドユーザーが定義するカテゴリ別ドキュメントやディスカッションのプロファイル項目と一致する関連情報の項目数を要約します。

検索サーバーを使用して、次のタイプのコンテンツを一致させることができます。

- 指定日数の範囲内でターゲットカテゴリに属する新規ドキュメント
- 指定日数の範囲内のディスカッションに含まれる新規関連コメント
- 保存済み検索に対するドキュメントのヒット

結果は、一致情報の数を示すリンクとしてプロファイルエントリに表示されます。エンドユーザーがこのリンクをクリックすると、その一致自体の詳細が表示されます。

カテゴリ登録では、エンドユーザーがリンクをクリックすると検索チャンネルが表示されます。ここには、興味対象の特定のドキュメントが標準のカテゴリ検索結果の形式で要約されます。登録チャンネルは、より詳細な情報への入り口として機能します。

プロファイラ機能により、指定した興味対象のコンテンツが変更された場合に、メールで通知できます。プロファイラは、Access Manager からエンドユーザーの登録情報を入手し、検索サーバーから結果を取り込んで、電子メール通知をエンドユーザーに送信します。プロファイラをスケジュール設定することで、組織レベルで特定の時間に実行できます。

登録のセットアップ

管理者は登録を有効または無効にすることができます。登録は次のレベルでセットアップ可能です。

- ルートレベル
- 組織レベル
- エンドユーザーレベル

▼ 登録をセットアップする

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「登録」タブをクリックします。
- 5 次のいずれかを選択して登録レベルを設定し、デフォルト値を設定します。
 - 「DNを選択」ドロップダウンメニューから「**TopLevel**[グローバル]」を選択します。

注-登録をトップレベルで管理する場合、システム全体を対象に、カテゴリ、ディスカッション、および保存済み検索の各タイプごとに登録のデフォルト最大数を設定します。

カテゴリ登録の最大数	ユーザーが登録できるカテゴリの最大数を指定します。
ディスカッション登録の最大数	ユーザーが登録できるディスカッションの最大数を指定します。
保存済み検索の最大数	保存できる検索の最大数を指定します。

- 「DNを選択」ドロップダウンメニューから任意の組織を選択します。

注-登録を組織レベルで管理する場合、システム全体を対象に、タイプ別(カテゴリ、ディスカッション、保存済み検索)の登録のデフォルト最大数を上書きします。

プロファイラ SMTP	電子メール通知をエンドユーザーに配信するSMTPサーバーとして動作するホストシステム。
プロファイラの電子メール	ユーザーが受信する電子メール通知の発信元となる登録プロファイラの電子メールアドレス。電子メールは、ID@domain の形式にしてください。
プロファイラプロバイダ	ユーザーへの電子メール通知の内容をレンダリングするために使用されるプロファイラチャンネルのURL。これは次の形式にしてください。 http://HOST:PORT/portal/dt?
プロファイラのデフォルト検索	デフォルトの検索サーバーのURL。プロファイラのデフォルト検索が使用されるのは、Portal Server 6.3.x によって作成されたユーザープロファイルとの下位互換を目的とする場合だけです。これは次の形式にしてください。 http://HOST:PORT/search1/search
プロファイラの最大ヒット数	ユーザーに送信された電子メール通知で見つかる組織内のエンドユーザー登録の結果ヒット数の最大値。たとえば、値が5の場合、「*」などの広範囲の保存済み検索は、もっとも関係のある5つの結果に制限されます。
カテゴリ登録の最大数	ユーザーが登録できるカテゴリの最大数。
ディスカッション登録の最大数	ユーザーが登録できるディスカッションの最大数。
保存済み検索の最大数	エンドユーザーが保存できる検索の最大数。

- 「DNを選択」ドロップダウンメニューから任意のユーザーを選択します。

注-組織ユーザーレベルで登録を管理する場合、ユーザーの登録設定を編集します。管理者は、ユーザーのサービスデータを管理できます。

- ユーザーの登録の更新
- ユーザーの登録の削除

プロファイラが有効になっている登録 「有効」を選択することで、ユーザーは電子メール通知を受け取ることができます。

登録のタイプごとに、登録を追加または消去します。形式は次のとおりです。

カテゴリ登録

label | target category | scope | lapsed time | rating | server | database | status

各表記の意味は次のとおりです。

label 編集される登録に与えられる論理参照で、文字列として指定する必要があります。必須フィールド。

target category ABC:DEF:GHI という形式の文字列で指定する必要があります。

scope 検索クエリー。検索演算子を含む有効な検索文字列として指定する必要があります。

lapsed time 次のいずれかの値を指定する必要があります。

- 0 = すべて
- 1 = 1 日以内
- 7 = 1 週間以内
- 30 = 1 ヶ月以内
- 180 = 6 ヶ月以内
- 365 = 1 年以内

rating これは、一致するドキュメントが登録の一致として選択される最小の重要度です。

値は次の数値です。

- -1 = 重要でない
- 0 = 普通
- 1 = 興味あり
- 2 = 重要

	<ul style="list-style-type: none"> ■ 3 = 必読
server	これは、登録の条件に一致する内容を見つける際にクエリー対象となる検索サーバーの URL です。
database	登録が一致を検索するターゲットの検索サーバーデータベース。これは単一値のデータベースです。
status	登録がアクティブであるかアクティブでないかを示すブール値。 <ul style="list-style-type: none"> ■ アクティブとは、登録が評価されることを意味します。 ■ アクティブでないとは、登録が休止していることを意味します。

ディスカッション登録

label | target discussion | scope | lapsed time | rating | server | database | status

各表記の意味は次のとおりです。

label	編集される登録に与えられる論理参照で、文字列として指定する必要があります。必須フィールド。
target discussion	登録がほかに定義された条件に対して一致する内容を検索する対象となるディスカッションスレッドの親ノード。
scope	検索クエリー。scope は、検索演算子を含む有効な検索文字列として指定する必要があります。
lapsed time	次のいずれかの値を指定する必要があります。 <ul style="list-style-type: none"> ■ 0 = すべて ■ 7 = 1 週間以内 ■ 30 = 1 ヶ月以内 ■ 180 = 6 ヶ月以内 ■ 365 = 1 年以内

rating	<p>これは、一致するドキュメントが登録の一致として選択される最小の重要度です。</p> <p>値は次の数値です。</p> <ul style="list-style-type: none"> ▪ -1 = 重要でない ▪ 0 = 普通 ▪ 1 = 興味あり ▪ 2 = 重要 ▪ 3 = 必読
server	<p>これは、登録の条件に一致する内容を見つける際にクエリー対象となる検索サーバーの URL です。</p>
database	<p>登録が一致を検索するターゲットの検索サーバーデータベース。これは単一値のデータベースです。</p>
status	<p>登録がアクティブであるかアクティブでないかを示すブール値。</p> <ul style="list-style-type: none"> ▪ アクティブとは、登録が評価されることを意味します。 ▪ アクティブでないとは、登録が休止していることを意味します。

保存済み検索

label | scope | lapsed time | rating | server | database | status

各表記の意味は次のとおりです。

label	<p>編集される登録に与えられる論理参照で、文字列として指定する必要があります。必須フィールド。</p>
scope	<p>検索クエリー。検索演算子を含む有効な検索文字列として指定する必要があります。</p>
lapsed time	<p>次のいずれかの値を指定する必要があります。</p> <ul style="list-style-type: none"> ▪ 0 = すべて

	<ul style="list-style-type: none"> ▪ 1 = 1 日以内 ▪ 7 = 1 週間以内 ▪ 30 = 1 ヶ月以内 ▪ 180 = 6 ヶ月以内 ▪ 365 = 1 年以内
rating	<p>これは、一致するドキュメントが登録の一致として選択される最小の重要度です。</p> <p>値は次の数値です。</p> <ul style="list-style-type: none"> ▪ -1 = 重要でない ▪ 0 = 普通 ▪ 1 = 興味あり ▪ 2 = 重要 ▪ 3 = 必読
server	これは、登録の条件に一致する内容を見つける際にクエリー対象となる検索サーバーの URL です。
database	登録が一致を検索するターゲットの検索サーバーデータベース。これは単一値のデータベースです。
status	<p>登録がアクティブであるかアクティブでないかを示すブール値。</p> <ul style="list-style-type: none"> ▪ アクティブとは、登録が評価されることを意味します。 ▪ アクティブでないとは、登録が休止していることを意味します。

6 「保存」をクリックします。

参考 同等の psadmin コマンド

```
psadmin set-attribute
```

Portal Server ディスカッションの管理

ここでは、ディスカッションチャンネルとその管理方法について説明します。

ここでは、次の内容について説明します。

- [122 ページの「DiscussionProvider の概要」](#)
- [123 ページの「DiscussionProvider の管理」](#)
- [125 ページの「DiscussionLite チャンネル」](#)

DiscussionProvider の概要

ディスカッションチャンネルは DiscussionProvider に基づいていて、検索チャンネルの JavaServer Pages™ (JSP™) ファイルに似ています。ディスカッションチャンネルにはクエリー部分と表示部分があります。また、デスクトップテーマを使用します。

DiscussionProvider の特徴は次のとおりです。

- デスクトップテーマを使用します
- JSP テクノロジーに基づいています
- 検索タグライブラリと API を使用して、バックエンドの検索サービスからデータを取得します

ディスカッションとコメントは、異なるリソースディスクリプタ (RD) としてディスカッションデータベースに格納されます。DiscussionProvider がサポートする内容は次のとおりです。

- 次の機能を持つ完全ビュー (ディスカッションチャンネルを使用) と省略ビュー (DiscussionLite チャンネルを使用)
- ディスカッションチャンネルから新しいディスカッションを開始する。
- 既存のディスカッションに返信する。
- 検索チャンネルからの Web ドキュメントに基づく新しいディスカッションを開始する。
- 次の機能を持つディスカッションリスト
 - 最終更新日でソートされた主要送信を取得する。
 - ユーザーが古いディスカッションにアクセスできるようにページ付けを行う。
- ディスカッションの各サブツリーを表示するディスカッション表示。主要項目は詳細に表示され、サブツリーは主要項目の下に表示されます。ここに表示される内容には、次の項目が含まれます。
 - ページ上の複数のフィルタ。ドキュメントの重要度 (「重要でない (irrelevant)」、「普通 (routine)」、「興味 (interesting)」、「重要 (important)」、「必読 (must read)」) などのフィルタに基づいてドキュメントを表示できます。
 - スレッド表示とフラット表示の表示設定を選択できます。
 - 展開しきい値を使用すると、サブツリー内の表示を制御しやすくなります。ユーザーは、評価の高いドキュメントだけを展開したり、すべてのドキュメントを展開したり閉じたりすることを選択できます。デフォルト値は「すべて縮小」です。「すべて展開」を選択した場合、フィルタ条件を満たすすべてのコメントが表示されます。また、ディスカッションの説明およびディスカッション評価用メニューが表示されるほか、返信を投稿する仕組みも提供されます。

- ディスカッション内の検索に対応しています。ユーザーは、チャンネル編集ページからこれらの設定を編集することもできます。
- ディスカッションに対してコメント追加し、重要度を設定する。たとえば、ユーザーは次の操作が可能です。
 - 既存のディスカッションにコメントを追加する。
 - すべてのディスカッションとコメントに重要度を設定する。ユーザーが設定する重要度はただちには適用されません。重要度の計算はアルゴリズムに基づいています。コメントの重要度は段階的に上下します。たとえば、コメントの重要度を「重要」に引き上げるには、「重要」という設定を3回行う必要があります。
 - ディスカッション内のすべてのディスカッションを検索する。これらの機能は、検索プロバイダにルーティングされます。ユーザーは、詳細検索機能を使用して、重要度による検索も実行できます。
 - 登録。認証されたユーザーは、登録のリンクを選択することで、特定のディスカッションを選択的に登録できます。この要求は、SubscriptionProvider によって処理されます。

DiscussionProvider の管理

DiscussionProvider チャンネルを作成し、ポータルサーバーの管理コンソールで管理できます。

- 123 ページの「DiscussionProvider からチャンネルを作成する」
- 124 ページの「DiscussionProvider チャンネルを削除する」
- 125 ページの「DiscussionProvider チャンネルを設定する」

エンドユーザーは、チャンネルの編集ページを使用して、ディスカッションチャンネルを設定できます。

▼ DiscussionProvider からチャンネルを作成する

- 1 [Portal Server 管理コンソールにログインします。](#)
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「DN を選択」ドロップダウンメニューから任意の DN を選択します。
- 5 チャンネルを作成するコンテナを選択します。
コンテナの「作業」と「プロパティ」が右側のパネルに表示されます。

- 6 「作業」の下の「新規チャンネルまたはコンテナ」をクリックしてウィザードを起動します。
 - a. 「ポータルを選択」ドロップダウンメニューからポータルサーバーを選択します。
 - b. 「DNを選択」ドロップダウンメニューから任意のDNを選択します。
 - c. 「タイプ」からチャンネルを選択して、「次へ」をクリックします。
 - d. 「チャンネルタイプ」から「プロバイダチャンネル」を選択し、「次へ」をクリックします。
 - e. 「プロバイダ」ドロップダウンメニューから「DiscussionProvider」を選択し、「次へ」をクリックします。
 - f. テキストボックスにチャンネルの名前を入力し、「次へ」をクリックします。
 - g. チャンネルの情報を確認し、「完了」をクリックします。
 - h. 「閉じる」をクリックします。DiscussionProvider に基づいたチャンネルが作成されます。

▼ DiscussionProvider チャンネルを削除する

- 1 [Portal Server 管理コンソールにログイン](#)します。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「DNを選択」ドロップダウンメニューから DiscussionProvider チャンネルが存在する DN を選択します。

ヒント- 「DP_ROOT」の下のチャンネルとコンテナをすべて一覧表示するには、このドロップダウンメニューから「DP XML ツリー」を「表示タイプ」として選択します。

- 5 チャンネルが存在するコンテナを選択します。
コンテナの「タスクおよびプロパティ」ページが表示されます。
- 6 「削除するチャンネルまたはコンテナを選択」をクリックします。

- 7 DiscussionProvider チャンネルを選択します。
- 8 「削除」をクリックします。

▼ DiscussionProvider チャンネルを設定する

- 1 Portal Server 管理コンソールにログインします。
- 2 「ポータル」タブを選択します。
- 3 「ポータル」からポータルサーバーを選択します。
- 4 「DN を選択」ドロップダウンメニューから DiscussionProvider チャンネルが存在する DN 組織を選択します。

ヒント- 「DP_ROOT」 の下のチャンネルとコンテナをすべて一覧表示するには、このドロップダウンメニューから「DP XML ツリー」を「表示タイプ」として選択します。

- 5 設定する DiscussionProvider チャンネルを選択します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

DiscussionLite チャンネル

DiscussionLite チャンネルには、最新のディスカッションタイトルと日付が20個表示されます。ディスカッションは、作成日(最終更新日)でソートされ、もっとも新しいディスカッションが先頭に表示されます。タイトルは、再設定可能です。

DiscussionLite チャンネルの表示には次のリンクがあります。

- 各ディスカッションの表示。
- ディスカッションチャンネルをターゲットとするすべてのディスカッションの表示。
- ディスカッションの開始。

デフォルトでは、チャンネルは単一のコンテナ内に表示され、すべてのリンクは JSPDynamicSingleContainer に持ち込まれます。

プロパティは、管理コンソールから設定できます。デフォルトでは、このチャンネルのプロパティをエンドユーザーが編集することはできません。

Portal Server シングルサインオンアダプタの管理

この章では、SSO(シングルサインオン)アダプタを設定して、エンドユーザーが利用できるオプションを調節する方法について説明します。この章で説明する内容は次のとおりです。

- 127 ページの「シングルサインオンアダプタの概要」
- 128 ページの「メタアダプタの管理」
- 130 ページの「アダプタの管理」
- 131 ページの「匿名ユーザーの作成」

シングルサインオンアダプタの概要

シングルサインオンアダプタサービスを使用することで、エンドユーザーは Portal Server プロバイダやその他の Web アプリケーションなどのアプリケーションを使用し、1回のサインオンだけで各種リソースサーバーに認証アクセスすることができます。アクセスできるリソースサーバーは、システムに実装されている SSO アダプタインタフェースによって異なります。

Portal Server は、次のリソースサーバーの SSO アダプタを提供しています。アドレス帳、カレンダー、およびメール。Instant Messaging チャネルのシングルサインオンは、SSO アダプタ経由では利用できませんが、Sun Java System Portal Server 認証方法経由で利用できます。この方法については、Instant Messaging チャネルの `authMethod` プロパティを参照してください。アドレス帳、カレンダー、メールの各サービスは、次の製品で利用できます。

- Sun Java System Calendar Server 5.1.1、6.0、6 2006Q2
- Sun Java System Sun Java System Messaging Server 5.2、6.0、6 2006Q2

リソースサーバーには、通常はメールサーバー用の JavaMail™ API など、標準的な API (アプリケーションプログラミングインタフェース) を使用して、アプリケーションによってアクセスされます。API を使用して認証接続を作成するには、API に

接続の設定データを提供する必要があります。SSO アダプタの目的は、この設定データを供給することであり、データの格納には SSO アダプタサービスが使用されます。

SSO アダプタサービスは、メタアダプタとアダプタという2つのレベルのデータを定義します。メタアダプタは、ユーザーが利用することになる接続のクラスを定義します。複数のユーザーが1つのメタアダプタを使用できます。メタアダプタは、そのメタアダプタを使用するすべてのユーザーに同一のデータ値を定義します。これには、デフォルト値と、ユーザーが編集できる値に関する情報が含まれます。このため、メタアダプタは、グローバルサービスレベルで定義されます。

アダプタは、メタアダプタの設定に加えて、組織、ロール、またはユーザーに固有のデータ値を提供します。アダプタはメタアダプタを参照し、ユーザーが編集できないプロパティのデータ値をメタアダプタから取得します。エンドユーザーが、アダプタの編集可能なプロパティを変更すると、そのアダプタは、そのエンドユーザーだけに適用されます。

SSO アダプタサービスを使用する Sun Java System Portal Server の通信チャンネルは、メタアダプタまたはアダプタのいずれかを参照し、リソースサーバーとの接続に必要なデータ値を取得します。チャンネルがメタアダプタを参照する場合、ユーザーが設定情報を保存すると、チャンネルはアダプタを参照ようになります。参照されたアダプタは、メタアダプタを参照します。

SSO アダプタの管理はすべて、Portal Server のコンソール Web アプリケーションか psadmin コマンド行インタフェースで行います。Portal Server コンソールのデフォルトの配備 URI は、/psconsole です。psadmin CLI のデフォルトの場所は、Solaris の場合、/opt/SUNWportal/bin です。

メタアダプタの管理

メタアダプタは、ユーザーが利用することになる接続のクラスを定義します。複数のユーザーが1つのメタアダプタを使用できます。

メタアダプタを使用して次のタスクを実行できます。

- [129 ページの「メタアダプタを表示する」](#)
- [129 ページの「メタアダプタを作成する」](#)
- [129 ページの「アダプタを表示する」](#)

▼ メタアダプタを表示する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「SSO アダプタ」タブを選択します。
メタアダプタのリストがテーブルに表示されます。

参考 同等の psadmin コマンド

```
psadmin list-ssoadapters
```

▼ メタアダプタを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「SSO アダプタ」タブを選択します。
- 3 「メタアダプタのリスト」から「新しいメタアダプタ」をクリックしてウィザードを起動します。
- 4 手順に従った後、「了解」をクリックし、指定されたメタアダプタを作成します。

参考 同等の psadmin コマンド

```
psadmin create-ssoadapter-template
```

▼ アダプタを表示する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「SSO アダプタ」タブを選択します。
 - DN のアダプタを表示するには、「アダプタを場所ごとに表示」をクリックします。
 - a. 「DN を選択」ドロップダウンメニューで任意の DN を選択します。
選択した DN のアダプタが一覧表示されます。

- メタアダプタのアダプタを表示するには、「メタアダプタのリスト」でメタアダプタを選択します。
 - a. 「選択したメタアダプタのアダプタを表示」をクリックします。

参考 同等の psadmin コマンド

```
psadmin list-ssoadapters
```

注 - CLI に許可されるアダプタのリストは、DN によるリストのみです。

アダプタの管理

アダプタは、メタアダプタの設定に加えて、組織、ロール、またはユーザーに固有のデータ値を提供します。アダプタはメタアダプタを参照し、ユーザーが編集できないプロパティのデータ値をメタアダプタから取得します。エンドユーザーが、アダプタの編集可能なプロパティを変更すると、そのアダプタは、そのエンドユーザーだけに適用されます。

SSO アダプタ設定を使用して次のタスクを実行できます。

- [130 ページの「アダプタを作成する」](#)
- [131 ページの「アダプタの設定プロパティを編集する」](#)

▼ アダプタを作成する

- 1 [Portal Server 管理コンソール](#)にログインします。
- 2 「SSO アダプタ」タブを選択します。
- 3 「メタアダプタのリスト」からメタアダプタを選択します。
- 4 「選択したメタアダプタのアダプタを表示」をクリックします。
- 5 「新しいアダプタ」をクリックします。
「新しいアダプタ」ページが表示されます。
- 6 必要に応じて設定属性を指定します。
- 7 「了解」をクリックします。

参考 同等の psadmin コマンド

```
create-ssoadapter-config
```

▼ アダプタの設定プロパティを編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「SSO アダプタ」タブを選択します。
- 3 「アダプタを場所ごとに表示」をクリックします。
- 4 「DN を選択」ドロップダウンメニューで任意の DN を選択します。
アダプタのリストが表示されます。
- 5 アダプタを選択し、必要に応じて設定属性を変更します。
- 6 「了解」をクリックします。

参考 同等の psadmin コマンド

```
psadmin set-ssoadapter-property
```

匿名ユーザーの作成

エンドユーザーは、管理者が設定した読み取り専用通信チャンネルにログインなしでアクセスすることができます。ただし、通常はエンドユーザーがこれらのチャンネルを編集することはできません。

▼ 匿名ユーザーのリストを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「SSO アダプタ」タブを選択します。
- 3 「SSO アダプタの作業」から「SSO アダプタに認証なしでアクセスできるユーザーのリストを編集」をクリックします。
- 4 「ユーザーの場所」から「ユーザーを追加」をクリックします。
- 5 「見つかったユーザー」テーブルからユーザーを選択します。

- 6 「選択したユーザーを追加」をクリックします。

注 - 匿名ユーザー機能は、Portal Server 管理コンソールでのみ使用できます。

パート II

検索サーバーの管理

- 第11章
- 第12章

検索サーバーの管理

この章では、Sun Java™ System Portal Server 検索サーバーの設定方法と管理方法について説明します。

この章で説明する内容は次のとおりです。

- 135 ページの「検索サーバーの概要」
- 137 ページの「検索サーバーの管理」
- 138 ページの「データベースの概要」
- 142 ページの「データベースの管理」
- 144 ページの「レポートの管理」
- 144 ページの「カテゴリの管理」

検索サーバーの概要

Portal Server 検索サーバーは、Google および Alta Vista などの一般的なインターネット検索サーバーと同様の検索インタフェースおよびブラウザインタフェースをサポートするために設計された分類およびデータベースサービスです。検索サーバーには、ドキュメントリソースを発見、変換、および要約するロボットが含まれます。Portal Server デスクトップには、JavaServer Pages™ (JSP™) に基づいた検索ユーザーインタフェースが含まれます。検索サーバーには、設定の編集を行うための管理ツールとシステム管理を行うためのコマンド行ツールが含まれます。設定は、Portal Server 管理コンソールを使用して定義および格納できます。

注-管理コンソールを使用すると、管理者は検索サーバーオプションの大部分を設定できますが、コマンド行インタフェースから使用できるすべての管理機能を実行できるわけではありません。

検索データベース

ユーザーは、検索サーバーのデータベースをクエリーして、リソースの場所を特定します。各データベースの個々のエントリーをリソース記述 (RD) と呼びます。リソース記述は、単一のリソースに関するサマリー情報を提供します。データベーススキーマは、各リソース記述のフィールドを決定します。

検索サーバーは、リソース記述メッセージ (RDM) やサマリーオブジェクト交換形式 (SOIF) など、インターネットのオープンスタンダードに基づいているため、クロスプラットフォームのエンタープライズ環境で動作できることが保証されています。

データベース分類カテゴリ

ユーザーが検索システムと対話する場合、2つの方法があります。クエリーを直接入力してデータベースを検索する方法と、カスタムなカテゴリセットを使用してデータベースコンテンツ経由で参照する方法です。カテゴリの階層構造は「分類」と呼ばれることがあります。リソースの分類は、データベースに目次を作成することにととえることができます。

ブラウズは、検索システムではオプションの機能です。つまり、カテゴリによるブラウズの機能がなくても、検索システムは十分有用なものになります。ユーザーで表示できるカテゴリを追加することがインデックスのユーザーに役立つかどうかを検討する必要があります。また、役立つ場合、作成するカテゴリの種類も検討する必要があります。

検索データベースのリソースは、検索手順を単純にするためにカテゴリに割り当てられます。データベースに多数の項目がある場合には、関連する項目をグループ化すると便利です。そのようにすると、ユーザーは特定の種類の項目をすばやく検索したり、類似する項目と比較したり、目的の項目を選択したりすることができます。

このような分類は製品およびサービスのインデックスに共通です。衣料品のカタログでは、衣類を男性用、女性用、および子供用に分類して、さらに、これらのカテゴリをコート、シャツ、靴などに分類します。オフィス製品カタログの場合、家具は文具、コンピュータ、およびソフトウェアと区別されます。広告ディレクトリは、製品とサービスのカテゴリによって編成されます。

印刷物の索引で項目をカテゴリ別に分類するときの原則は、オンラインインデックスにも当てはまります。この発想では、種類別に検索することが容易になるので、ユーザーは目的のリソースを選択できます。設計するインデックスの適用範囲を問わず、カテゴリを設定する際にもっとも重要なのは使いやすさです。ユーザーがどのようにカテゴリを使用するかを知る必要があります。たとえば、異なる場所に3つの支社を持つ企業のためにインデックスを設計する場合、トップレベルのカテゴリを支社ごとに作成しようとするかもしれません。ユーザーにとって、支社の所在地よりも各部門の機能の方が重要であれば、リソースを部門別に分類することが適切です。

カテゴリを定義すると、カテゴリにリソースを割り当てるためのルールを設定する必要があります。これらのルールを、「分類ルール」と呼びます。分類ルールを適切に定義しないと、ユーザーがカテゴリでブラウズしようとしても、リソースを検索できません。リソースは正しく分類する必要がありますが、同時にドキュメントの分類も適切でなければなりません。

検索サーバーの管理

Sun Java System Portal Server は 1 つ以上の検索サーバーをサポートします。

- [137 ページの「検索サーバーを作成する」](#)
- [138 ページの「検索サーバーを削除する」](#)

▼ 検索サーバーを作成する

Portal Server のインストール時にデフォルトの検索サーバー (*search1*) が作成されます。新規検索サーバーウィザードを使用して、新しい検索サーバーを作成することもできます。

始める前に 使用する Web コンテナインスタンスに固有の設定情報を認識しておく必要があります。

- Sun Java System Web Server 7
- Sun Java System Web Server 6
- Sun Java System Application Server 8.1
- BEA Weblogic 8
- IBM WebSphere 5

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」を選択し、メニューバーから「新規」を選択します。
新しい検索サーバーウィザードが表示されます。
- 3 指示に従い、「完了」をクリックして指定した検索サーバーを作成します。

参考 同等の `psadmin` コマンド

```
psadmin create-search-server。
```

▼ 検索サーバーを削除する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択します。
- 3 検索サーバーを選択し、「削除」をクリックします。

参考 同等の psadmin コマンド

```
psadmin delete-search-server
```

データベースの概要

検索サーバーはデータベースにリソース記述を格納します。検索データベースはドキュメントコレクションのインデックスです。これらは、インデクサ(コマンド `rdmgr`、または検索サーバー自体)によって作成されます。たとえばデフォルトでは、Web サイトとロボットインデックスを巡回し、見つけ出したものをユーザーがデータを検索できるデフォルトの検索データベースにすべて格納するようにロボットをセットアップできます。データやインデックスはほかのデータベースに格納されます。

データベース管理のために行う必要のある設定作業およびメンテナンス作業は次のとおりです。

- 138 ページの「データベースへのインポート」
- 139 ページの「データベーススキーマの編集」
- 140 ページの「スキーマのエイリアスの定義」
- 140 ページの「データベース分析の表示」
- 140 ページの「データベースインデックスの再作成」
- 140 ページの「データベースの期限切れ」
- 141 ページの「データベースの破棄」
- 141 ページの「データベースのパーティションの作成」

データベースへのインポート

通常、検索データベースの項目はロボットから取得します。また、ほかの Portal Server 検索エンジン、iPlanet Web Server や Netscape™ Enterprise Server、または別のソースから生成したデータベースから既存の項目のデータベースをインポートすることもできます。ロボットを使用して新規に作成する代わりに RD の既存データベースをインポートすると、ネットワークトラフィックが減少します。また、大量のインデックス作成作業を分割できるので、より高速にインデックスを作成できます。

中央データベースがインデックスを作成するサーバーから物理的に遠くにある場合には、RDをローカルに生成して、中央データベースに各地点のリモートデータベースを定期的にインポートすると便利です。

検索サーバーは、インポートエージェントを使用して、別のサーバーやデータベースからRDをインポートします。「インポートエージェント」は、外部ソースからRDを検出して、ローカルデータベースに情報をマージするプロセスです。

データベースをインポートする前に、インポートエージェントを作成する必要があります。エージェントを作成したら、すぐにインポートプロセスを開始することも、インポートプロセスが定期的に行われるようにスケジュールを設定することもできます。

データベーススキーマの編集

「スキーマ」は、検索サーバーで管理するリソースごとの情報とその形式を決定します。スキーマの設計によって、インデックスのユーザビリティに影響する次の2つの要因が決まります。

- ユーザーがリソースを検索する方法
- ユーザーがリソース情報を表示する方法

スキーマはデータベースのリソース記述のマスタデータ構造です。データ構造フィールドのインデックス作成方法と定義方法によって、リソースへのアクセス頻度が変わります。

スキーマは、検索サーバーとそのロボットが使用するファイルの構造に密接に結びついています。管理コンソールでスキーマツールを使用して、データ構造だけを変更してください。スキーマファイルは直接編集しないでください。

検索サーバーのデータベーススキーマを編集すると、新規のスキーマ属性の追加、スキーマ属性の変更、あるいは属性の削除ができます。

スキーマには次の属性があります。

- 編集可能: オンの場合、この属性はリソース記述エディタに表示される属性を示すので、ユーザーは値を変更できます。
- インデックス作成可能: この属性はユーザーがこのフィールド内の値を検索できることを示します。インデックス作成可能フィールドは、「詳細検索」画面のポップアップメニューに表示されることもあります。
- 説明: この属性はスキーマの記述に使用するテキスト文字列です。コメントや注釈に使用します。
- エイリアス: この属性は、インポートしたデータベーススキーマ名をユーザー独自のスキーマに変換するためのエイリアスを定義します。
- 乗数のスコア: 特定の要素をスコアリングする重み付けフィールドです。有効値は正の数です。

- データタイプ: データタイプを定義します。

スキーマのエイリアスの定義

データベーススキーマのフィールド名が互いに矛盾していることがあります。あるサーバーから別のサーバーにリソース記述をインポートするとき、2つのサーバーのスキーマの項目名が同一であることを、常に保証することはできません。同様に、ロボットがHTML <meta> タグをドキュメントからスキーマフィールドに変換するとき、名前はドキュメントによって制御されます。

検索サーバーは、ユーザーのスキーマ属性にスキーマのエイリアスを定義して、外部スキーマ名をデータベースのフィールドに有効な名前にもマッピングできるようにします。

データベース分析の表示

検索サーバーのレポートには、インデックスを作成したサイトの数、およびデータベースの各サイトのリソースの数などの情報が含まれます。

データベースインデックスの再作成

スキーマを編集してインデックスフィールドを追加または削除する場合、またはディスクエラーによってインデックスファイルが破損した場合は、検索サーバーが使用するリソース記述データベースのインデックスを再作成する必要があります。また、別の何らかの理由でデータベースのコンテンツとインデックスの間に矛盾が発生すると、再作成が必要となることがあります。たとえば、インデックス作成中にシステム障害が発生した場合です。

大規模なデータベースのインデックスを再作成するのに、数時間を要する場合があります。データベースのインデックスの再作成に要する時間は、データベースに含まれるレコードの数に応じて変わります。大規模なデータベースでは、サーバーの負荷が低い時にインデックスの再作成を行うようにしてください。

データベースの期限切れ

データベースを「期限切れ」にすると、古くなったリソース記述も削除されます。リソース記述が削除されるのは、期限切れを実行したとき「だけ」です。期限切れのリソース記述が削除されても、データベースのサイズは減少しません。

リソース記述の属性の1つに有効期限があります。ロボットはHTML <meta> タグ、またはリソースのサーバーが提供する情報を基に有効期限を設定します。リソース

が有効期限を指定しないかぎり、リソース記述はデフォルトで作成から3カ月有効です。検索サーバーで検索データベースから期限切れのリソース記述を定期的に破棄するようにしてください。

データベースの破棄

破棄を実行すると、データベースの中身を削除できます。これによって、インデックス用のディスク領域は回復しますが、メインデータベースのディスク領域は回復できません。これは、この領域が新規のデータをデータベースに追加したときに再利用されるためです。

データベースのパーティションの作成

検索サーバーを使用すれば、複数のディスク、ファイルシステム、ディレクトリ、またはパーティションにまたがる検索データベースを構成する物理ファイルを配置できます。異なる物理デバイス、または論理デバイスにデータベースを拡張することで、単一のデバイスに使用するよりも大きなデータベースを作成することができます。

デフォルトでは、検索サーバーは1つのディレクトリだけを使用するデータベースを設定します。コマンド行インタフェースを使用すると、データベースパーティションに次の2種類の操作を実行できます。

- 新しいパーティションの追加
- パーティションの移動

検索サーバーは、各パーティションに空き領域があるかどうかを確認しません。データベースに適切な空き領域を確保するのは、ユーザーの作業です。

データベースのパーティションは、合計で最大15個追加できます。

注—一度パーティションの数を増やすと、あとでパーティションの数を減らすには、データベース全体を削除する必要があるので注意してください。

ただし、ディスク容量が十分にある場合に限り、パーティションの作成はお勧めしません。

データベースパーティションの物理的な場所を変更するには、新しい場所の名前を指定します。同様に、既存のパーティションの名前も変更できます。パーティションを操作するには `rdmgr` コマンドを使用します。 `psadmin` コマンドの詳細については、『Sun Java System Portal Server 7.1 Command Line Reference』を参照してください。

データベースの管理

データベースを管理するには、次の手順に従います。

- 142 ページの「データベースを作成する」
- 142 ページの「インポートエージェントを作成する」
- 143 ページの「リソース記述を作成する」
- 143 ページの「リソース記述を管理する」

▼ データベースを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「データベース」をクリックし、メニューバーの「管理」をクリックします。
- 4 「新規」をクリックします。
「新しいデータベース」ページが表示されます。
- 5 新しいデータベースの名前を入力し、「了解」をクリックします。

参考 同等の psadmin コマンド

```
psadmin create-search-database
```

▼ インポートエージェントを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「データベース」をクリックし、メニューバーの「エージェントのインポート」をクリックします。
- 4 「新規」をクリックしてウィザードを起動します。
- 5 「エージェントのインポート」属性を指定します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Import Agents」を参照してください。
- 6 「終了」をクリックします。

参考 同等の psadmin コマンド

```
psadmin create-search-importagent
```

▼ リソース記述を作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「データベース」をクリックし、メニューバーの「管理」をクリックします。
- 4 データベースを選択し、「リソース記述を管理」をクリックします。
- 5 「新規」をクリックし、属性を指定します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Schema」を参照してください。
- 6 「了解」をクリックします。

▼ リソース記述を管理する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「データベース」をクリックし、メニューバーの「管理」をクリックします。
- 4 データベースを選択し、「リソース記述を管理」をクリックします。
- 5 リソース記述を選択して、次のいずれかの操作を実行します。
 - 編集
 - すべて編集
 - 削除

属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Schema」を参照してください。

- 6 「保存」をクリックします。

参考 同等の psadmin コマンド

```
psadmin modify-search-resourcedescription
```

レポートの管理

検索サーバーには、検索アクティビティを監視できる多数のレポートが用意されています。

▼ レポートを表示する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 メニューバーの「レポート」をクリックします。
- 4 メニューバーのリンクを選択すると特定のレポートが表示されます。
次のオプションを使用できます。
 - ログ
 - ロボットの詳細レポート
 - 頻度の高い検索
 - 除外 URL

カテゴリの管理

次の作業に従って、カテゴリを管理できます。

- 144 ページの「カテゴリを作成する」
- 145 ページの「カテゴリを編集する」
- 145 ページの「自動分類を実行する」
- 145 ページの「自動分類の属性を編集する」

▼ カテゴリを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「カテゴリ」を選択し、メニューバーの「参照/検索」を選択します。

- 4 「新規」をクリックします。
「新しい検索カテゴリ」ダイアログが表示されます。
- 5 必要に応じて属性を指定します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Manage Categories」を参照してください。
- 6 「了解」をクリックします。

▼ カテゴリを編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「カテゴリ」をクリックし、メニューバーの「参照/検索」を選択します。
- 4 カテゴリを選択し、「編集を」をクリックして「*Category*の編集」ページを表示します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Manage Categories」を参照してください。

▼ 自動分類を実行する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブを選択し、検索サーバーを選択します。
- 3 「カテゴリ」をクリックし、メニューバーの「自動分類」を選択します。
- 4 「自動分類を実行」をクリックします。

▼ 自動分類の属性を編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 「検索サーバー」タブをクリックし、検索サーバーを選択します。
- 3 「カテゴリ」をクリックし、メニューバーの「自動分類」を選択します。

- 4 必要に応じて属性を変更します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。
- 5 「保存」をクリックします。

検索サーバーロボットの管理

この章では、Sun Java™ System Portal Server 検索サーバーロボットとその対応する設定ファイルについて説明します。この章には、次のトピックが含まれています。

- 147 ページの「検索サーバーロボットについて」
- 152 ページの「ロボットの管理」
- 155 ページの「リソースフィルタリング処理」
- 159 ページの「フィルタの管理」
- 161 ページの「分類ルールの管理」
- 179 ページの「変更可能なプロパティ」
- 186 ページの「robot.conf ファイルの例」
- 162 ページの「ソースとデスティネーション」
- 165 ページの「セットアップ関数」
- 166 ページの「フィルタリング関数」
- 169 ページの「サポート関数のフィルタリング」
- 175 ページの「列挙関数」
- 176 ページの「生成関数」
- 178 ページの「シャットダウン関数」

検索サーバーロボットについて

検索サーバーロボットは、ドメイン内のリソースを特定し、レポートを作成するエージェントです。これには、2種類のフィルタを使用します。列挙子フィルタとジェネレータフィルタです。

列挙子フィルタは、ネットワークプロトコルを使用してリソースを検出します。列挙子フィルタは、各リソースをテストし、適切な基準に一致した場合はリソースを列挙します。たとえば、列挙子フィルタはHTMLファイルからハイパーテキストリンクを抽出し、そのリンクを使用して別のリソースを検索できます。

ジェネレータフィルタは各リソースをテストし、リソース記述 (RD) を作成する必要があるかどうかを確認します。リソースがテストに合格した場合、ジェネレータは検索サーバーデータベースに格納される RD を作成します。

ロボットを管理するために必要な設定と保守タスクについては、次の節で説明します。

- [150 ページの「サイトの定義」](#)
- [150 ページの「ロボットの巡回の制御」](#)
- [151 ページの「ロボットユーティリティーの使用」](#)
- [152 ページの「ロボットのスケジュール設定」](#)

ロボットの動作の仕組み

[図 12-1](#) は、ロボットが URL と関連するネットワークリソースを調べる方法を示しています。列挙機能と生成機能の両方が各リソースをテストします。リソースが列挙テストに合格すると、ロボットがそのリソースをほかの URL について検査します。リソースがジェネレータテストに合格すると、ロボットは検索サーバーデータベースに格納されているリソース記述を生成します。

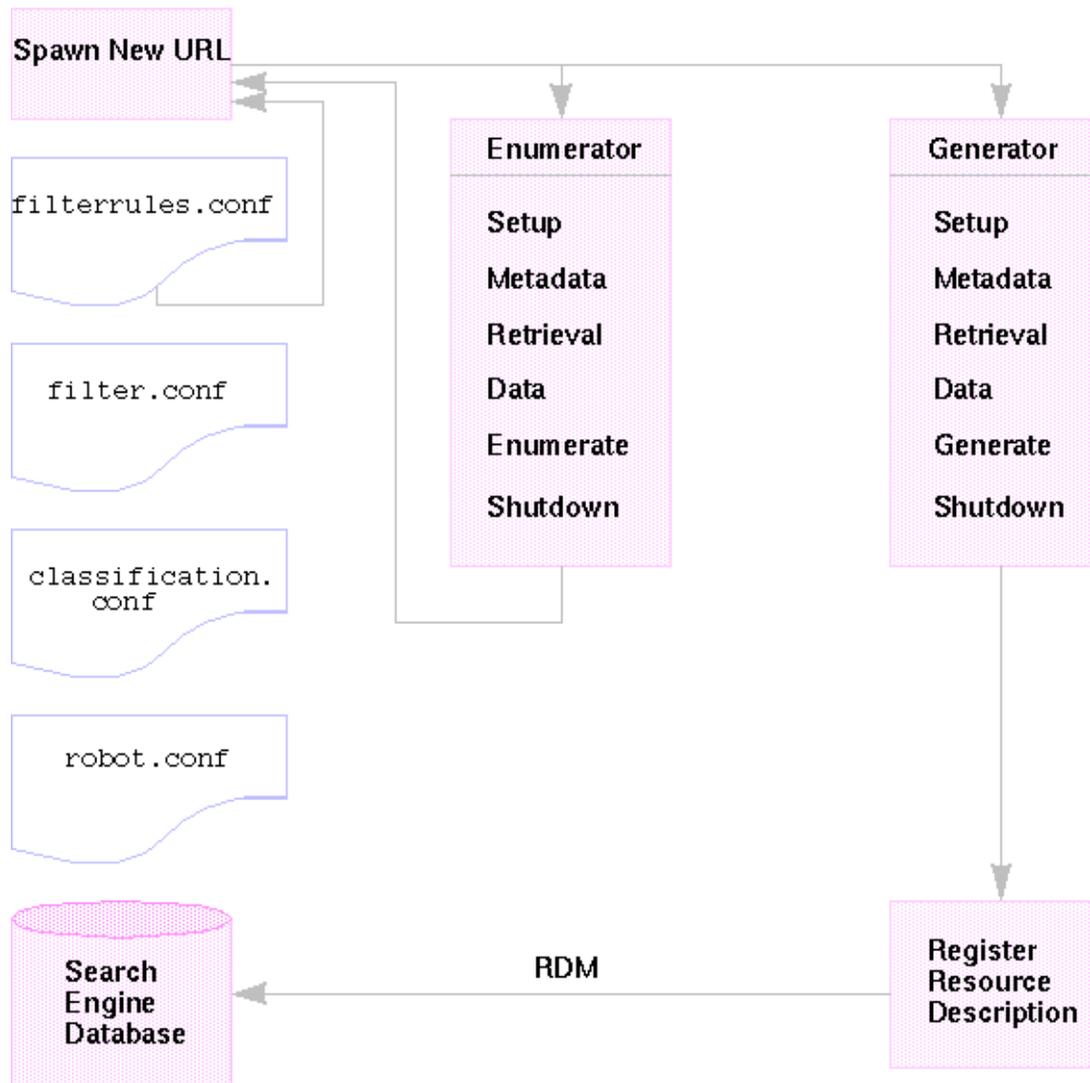


図12-1 ロボットの動作の仕組み

ロボット設定ファイル

ロボット設定ファイルでは、ロボットの動作が定義されます。これらのファイルは、`/var/opt/SUNWportal/searchservers/searchserverid/config` ディレクトリにあります。次のリストは、各ロボット設定ファイルを説明しています。

classification.conf	ロボットが生成するRDの分類に使用されるルールを含みます。
filter.conf	ロボットが使用する列挙フィルタと生成フィルタを定義します。
filterrules.conf	ロボットのサイト定義、開始ポイントURL、MIMEタイプに基づくフィルタリングルール、およびURLパターンを含みます。
robot.conf	ロボットのオペレーティングプロパティのほとんどを定義します。

検索サーバー管理インターフェースを使用すれば、ほとんどのプロパティを設定できるので、通常は `robot.conf` ファイルを編集する必要はありません。ただし、上級ユーザーはこのファイルを手動で編集することにより、インターフェースを介してでは設定できないプロパティを設定することができます。

サイトの定義

ロボットはリソースを見つけて、それらの記述をデータベースに追加するかどうかを決定します。どのサーバーを訪問し、サーバーのどの部分のインデックスを作成するかの判断は、サイト定義と呼ばれます。

ロボットのサイトの定義は、サーバー管理者のもっとも重要な作業の1つです。ロボットは、インデックス作成が必要なすべてのサーバーに送信しなければなりません。同時に、データベース領域を過度に拡張して正確な情報の検索を困難にする可能性のある外部サイトを排除することも必要です。

ロボットの巡回の制御

ロボットは、インデックスの作成のために選択したさまざまなサイトへのリンクを抽出し、追跡します。システム管理者は、次のようなさまざまな設定によって、これらのプロセスを制御できます。

- ロボットの起動、停止、スケジューリング
- ロボットが訪問するサイトの定義
- 巡回の頻度を指定する巡回属性
- フィルタを定義することによる、ロボットがインデックスを作成するリソースタイプ
- インデックス属性を定義することによる、ロボットがデータベースに作成するエントリの種類

ロボットの巡回属性についての説明は、『Sun Java System Portal Server 7.1 Technical Reference』を参照してください。

ロボットデータのフィルタリング

フィルタを使用すると、リソースの属性をフィルタ定義と比較することでリソースを識別できるようになるので、これを除外することも、含むこともできます。ロボットは多くの事前定義されたフィルタを提供します。一部のフィルタはデフォルトで有効です。次のフィルタは事前定義されています。アスタリスク(*)の付いたファイルがデフォルトで有効です。

- アーカイブファイル*
- オーディオファイル*
- バックアップファイル*
- バイナリファイル*
- CGI ファイル*
- 画像ファイル*
- Java、JavaScript、スタイルシートファイル*
- ログファイル*
- Lotus Domino ドキュメント
- Lotus Domino オープンビュー
- プラグインファイル
- Power Point ファイル
- リビジョン管理ファイル*
- ソースコードファイル*
- スプレッドシートファイル
- システムディレクトリ (UNIX)
- システムディレクトリ (NT)
- 一時ファイル*
- ビデオファイル*

新規フィルタ定義の作成、フィルタ定義の変更、またはフィルタの有効化または無効化が可能です。詳細は、155 ページの「リソースフィルタリング処理」を参照してください。

ロボットユーティリティーの使用

ロボットには次の2つのデバッグツールまたはユーティリティーがあります。

- サイトプローブ - DNS エイリアス、サーバーリダイレクト、仮想サーバーなどを確認します

- シミュレータ - 特定の URL でロボットフィルタリングを部分的にシミュレートします。シミュレータは表示されたサイトをロボットが受け入れるかどうかを示します。

ロボットのスケジュール設定

検索データを最新に維持するには、ロボットがサイトを定期的に検索して、インデックスを作成する必要があります。ロボットの巡回とインデックス作成は処理リソースとネットワークの帯域幅を消費する場合があるため、ピーク以外の時間帯にロボットを実行するようにスケジュール設定してください。管理コンソールを使用すると、管理者はロボットを実行するスケジュールをセットアップすることができます。

ロボットの管理

ここでは、ロボットを管理するための次のタスクについて説明します。

- 152 ページの「ロボットを起動する」
- 153 ページの「ロボットデータベースをクリアする」
- 153 ページの「サイト定義を作成する」
- 153 ページの「サイト定義を編集する」
- 154 ページの「ロボットの巡回とインデックス作成を制御する」
- 154 ページの「シミュレータを実行する」
- 155 ページの「サイトプローブユーティリティーを実行する」

▼ ロボットを起動する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択します。サーバーのリストから検索サーバーを選択します。
- 3 メニューバーの「ロボット」をクリックし、メニューの「状態と制御」をクリックします。
- 4 「開始」をクリックします。

参考 同等の psadmin コマンドの場合

```
psadmin start-robot
```

注 - コマンド `psadmin start-robot` の場合、ロボットが巡回できるサイトが定義されていない場合、検索ロボットは起動しません。コマンド `psadmin start-robot` は、サイトが使用できない場合、次のように表示します。開始位置:0 個が定義されています。

▼ ロボットデータベースをクリアする

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーの「ロボット」を選択し、「状態と制御」を選択します。
- 4 「ロボットデータベースをクリア」をクリックします。

▼ サイト定義を作成する

ロボットはリソースを見つけて、それらの記述をデータベースに追加するかどうかを決定します。どのサーバーを訪問し、サーバーのどの部分のインデックスを作成するかの判断は、サイト定義と呼ばれます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「サイト」を選択します。
- 4 「サイトの管理」の「新規」をクリックし、サイトの設定属性を指定します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Sites」を参照してください。
- 5 「了解」をクリックします。

▼ サイト定義を編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。

- 3 メニューバーから「ロボット」、「サイト」をクリックします。
- 4 変更するサイトの名前をクリックします。
「サイトを編集」ダイアログが表示されます。
- 5 必要に応じて設定属性を変更します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Sites」を参照してください。
- 6 「了解」をクリックして変更を記録します。

▼ ロボットの巡回とインデックス作成を制御する

ロボットは、インデックスの作成のために選択したさまざまなサイトを巡回します。巡回およびインデックス作成の操作プロパティを定義することでロボットがサイトを巡回する方法を制御できます。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「プロパティ」をクリックします。
- 4 必要に応じてロボットの巡回とインデックス作成の属性を指定します。
属性の詳細については、『Sun Java System Portal Server 7.1 Technical Reference』の「Site Probe」を参照してください。
- 5 「保存」をクリックします。

▼ シミュレータを実行する

シミュレータは、一覧表示された1つまたは複数のサイトのロボットフィルタリングの部分的なシミュレーションを実行します。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「ユーティリティー」をクリックします。

- 4 シミュレーションを行う新しいサイトの URL を「新しい URL を追加」テキストボックスに入力して「追加」をクリックします。
「既存のロボットサイト」の下に一覧表示される既存のサイト上でシミュレータを実行することもできます。
- 5 「シミュレータの実行」をクリックします。

▼ サイトプローブユーティリティーを実行する

サイトプローブユーティリティーは、DNS エイリアス、サーバーリダイレクト、仮想サーバーなどの情報を確認します。

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「ユーティリティー」をクリックします。
- 4 プローブするサイトの URL を入力します。
- 5 (省略可能) プローブで DNS 情報を取得するには、「サイトプローブ」で「DNS 情報の詳細を表示」を選択します。
- 6 「サイトプローブの実行」をクリックします。

リソースフィルタリング処理

ロボットはフィルタを使用して、処理するリソースとそのリソースの処理方法を決定します。ロボットがリソースと同時にリソースの参照を検出すると、各リソースにフィルタを適用します。フィルタはそれらのリソースを列挙し、検索サーバーデータベースに格納するリソース記述を生成するかどうかを決定します。

ロボットは1つあるいは複数の開始ポイント URL を調べ、フィルタを適用し、それらの URL などを列挙して生成された URL にそのフィルタを適用します。開始ポイント URL は `filterrules.conf` ファイルで定義されます。

列挙フィルタと生成フィルタはそれぞれ、必須である初期化操作を実行し、現在のリソースに比較テストを適用します。各テストは、リソースを許可または拒否することが目標です。各フィルタにはシャットダウンフェーズも備わっており、クリーンアップ操作を実行します。

リソースが許可されると、フィルタの通過が続行されます。最終的にロボットはそのリソースを列挙し、さらにほかのリソースを検出しようとします。ジェネレータはリソース記述も作成できます。

リソースが否認される場合、そのリソースは拒否されます。拒否されたリソースのフィルタは、それ以上動作しません。

これらの操作は必ずしも連携されていません。リソースには列挙につながるものもあれば、RD 生成につながるものもあります。多くのリソースは列挙にも、RD 生成にもつながります。たとえば、リソースが FTP ディレクトリである場合、一般的にはそのリソース用に RD は生成されません。ただし、ロボットは FTP ディレクトリの個別ファイルを列挙できます。ほかのドキュメントへのリンクを含む HTML ドキュメントは、RD を生成し、また、関連のドキュメントを列挙することもできます。

次の節では、フィルタ処理について説明します。

- 156 ページの「フィルタ処理の段階」
- 157 ページの「フィルタの構文」
- 158 ページの「フィルタディレクティブ」
- 159 ページの「フィルタの記述または修正」

フィルタ処理の段階

列挙フィルタ、生成フィルタのどちらにも、フィルタリング処理における 5 つのフェーズがあります。

- セットアップ: 初期化処理を行います。ロボットが存在する間、一度だけ発生します。
- メタデータ: そのリソースについて利用可能なメタデータに基づき、リソースをフィルタリングします。リソースがネットワークを介して取得される前に、メタデータのフィルタリングがリソースごとに 1 回実行されます。表 12-1 は、共通のメタデータタイプの例を示しています。

表 12-1 共通のメタデータタイプ

メタデータタイプ	説明	例
完全な URL	リソースの場所	http://home.siroe.com/
プロトコル	URL のアクセス部分	http, ftp, file
ホスト	URL のアドレス部分	www.siroe.com
IP アドレス	ホストの数値バージョン	198.95.249.6

表 12-1 共通のメタデータタイプ (続き)

メタデータタイプ	説明	例
PATH	URL のパス部分	/index.html
深さ	開始ポイント URL からのリンク数	5

- データ: リソースのデータに基づいてリソースをフィルタリングします。データのフィルタリングは、ネットワークを介してデータが取得されたあとにリソースごとに1回実行されます。フィルタリングに使用できるデータには、次のものがあります。
 - content-type
 - content-length
 - content-encoding
 - content-charset
 - last-modified
 - expires
- 列挙: 検査すべきほかのリソースを参照するかどうかを判断するため、現行のリソース中の参照を列挙します。
- 生成: リソースのリソース記述 (RD) を生成し、検索サーバー データベースにその RD を保存します。
- シャットダウン: 必要な終了操作を実行します。この処理は、ロボットが存在する間、一度だけ発生します。

フィルタの構文

filter.conf ファイルには、列挙フィルタおよび生成フィルタの定義が含まれています。このファイルには、列挙と生成の両方について、複数のフィルタを含むことができます。ロボットが使用するフィルタは、robot.conf ファイルの enumeration-filter プロパティおよび generation-filter プロパティで指定されます。

フィルタの定義には、次のように明確に定義された構造があります。ヘッダー、本文、終了です。ヘッダーは、フィルタの開始を識別し、たとえば次のように名前を宣言します。

```
<Filter name="myFilter">
```

本文は、セットアップ、テスト、列挙または生成、およびシャットダウン中のフィルタの動作を定義する一連のフィルタディレクティブで構成されています。各ディレクティブでは、関数とその関数のプロパティ (該当する場合) を指定します。

終了は </Filter> によって示されます。

例 12-1 は、enumeration1 という名のフィルタを示しています。

例 12-1 列挙ファイルの構文

```
<Filter name="enumeration1">
  Setup fn=filterrules-setup config=./config/filterrules.conf
# Process the rules
  MetaData fn=filterrules-process
# Filter by type and process rules again
  Data fn=assign-source dst=type src=content-type
  Data fn=filterrules-process
# Perform the enumeration on HTML only
  Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
# Cleanup
  Shutdown fn=filterrules-shutdown
</Filter>
```

フィルタディレクティブ

フィルタディレクティブは、ロボットアプリケーション関数 (RAF) を使用して、操作を実行します。このディレクティブの使用法および実行の流れは、Sun Java System Web Server の `obj.conf` ファイルの NSAPI ディレクティブおよびサーバーアプリケーション関数 (SAF) とよく似ています。NSAPI、SAF と同様に、*pblocks* とも呼ばれるプロパティブロックを使用して、データは格納および送信されます。

6つのロボットディレクティブ、またはRAFクラスが、155ページの「リソースフィルタリング処理」に示したフィルタリングのフェーズや操作に対応しています。

- セットアップ
- メタデータ
- データ
- 列挙
- 生成
- シャットダウン

各ディレクティブは、独自のロボットアプリケーション関数を持っています。たとえば、Metadata ディレクティブおよびData ディレクティブではフィルタリング関数、Enumerate ディレクティブでは列挙関数、Generate ディレクティブでは生成関数を使用します。

ロボットアプリケーションの組み込み関数と、独自のロボットアプリケーション関数を記述する手順については、『Sun Java System Portal Server 7.1 Developer's Guide』を参照してください。

フィルタの記述または修正

ほとんどの場合、管理コンソールを使用すると、ほとんどのサイト定義ベースのフィルタを作成できます。修正が必要な場合は、`filter.conf` ファイルおよび `filterrules.conf` ファイルを変更することができます。これらのファイルは、`/var/opt/SUNWportal/searchservers/searchserverid/config` ディレクトリにあります。

より複雑なプロパティのセットを作成するには、ロボットが使用する設定ファイルを編集します。

フィルタを記述または変更する場合は、次の順序に注意してください。

- ディレクティブの実行 (特に各フェーズで使用可能な情報)
- `filterrules.conf` のフィルタルール

次も実行できます。

- `robot.conf` ファイルでプロパティを変更する。
- `filter.conf` ファイルでロボットアプリケーション関数を変更する。
- 独自のロボットアプリケーション関数を作成する。

詳細については、『[Sun Java System Portal Server 7.1 Developer's Guide](#)』を参照してください。

フィルタの管理

この節では、ロボットフィルタを管理するための次のタスクについて説明します。

- [159 ページの「フィルタを作成する」](#)
- [160 ページの「フィルタを削除する」](#)
- [160 ページの「フィルタを編集する」](#)
- [161 ページの「フィルタを有効または無効にする」](#)

▼ フィルタを作成する

- 1 [Portal Server 管理コンソールにログイン](#)します。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「フィルタ」を選択します。
- 4 「新規」をクリックします。
「新しいロボットフィルタウィザード」が表示されます。

- 5 指示に従って指定されたフィルタを作成します。
 - a. テキストボックスにフィルタ名とフィルタの説明を入力し、「次へ」をクリックします。
 - b. フィルタの定義と動作を指定し、「完了」をクリックします。
フィルタ属性の詳細については、「Filters」を『Sun Java System Portal Server 7.1 Technical Reference』で参照してください。
 - c. 「閉じる」をクリックして新しいフィルタを読み込みます。

▼ フィルタを削除する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「フィルタ」を選択します。
- 4 フィルタを選択します。
- 5 「削除」をクリックします。
- 6 表示される確認ダイアログボックスで「了解」をクリックします。

▼ フィルタを編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「フィルタ」を選択します。
- 4 フィルタを選択して「編集」をクリックします。
「フィルタを編集」ページが表示されます。
- 5 必要に応じて設定属性を変更します。
フィルタ属性の詳細については、「Filters」を『Sun Java System Portal Server 7.1 Technical Reference』で参照してください。
- 6 「了解」をクリックします。

▼ フィルタを有効または無効にする

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」、「フィルタ」を選択します。
- 4 フィルタを選択します。
 - フィルタを有効にするには、「有効」をクリックします。
 - フィルタを無効にするには、「無効」をクリックします。

分類ルールの管理

ドキュメントは、この値に設定した最大数を上限として、任意の数のカテゴリに割り当てることができます。分類ルールはフロー制御による判定を実行しないので、ロボットフィルタルールよりも単純です。分類ルールでは、特定のカテゴリをリソース記述の一部としてリソースに割り当てるための条件を決定する必要があります。分類ルールは、「もしある条件が真ならば、リソースを <カテゴリ> に割り当てる」という形式の簡単な条件文で作成されます。

▼ 分類ルールを作成する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 メニューバーから「ロボット」を選択し、「分類ルール」を選択します。
- 4 「分類ルール」を選択して「新規」をクリックします。
「新規分類ルール」ダイアログボックスが表示されます。
- 5 必要に応じて設定属性を指定します。
属性の詳細については、「Manage Classification Rules」を『Sun Java System Portal Server 7.1 Technical Reference』で参照してください。
- 6 「了解」をクリックします。

▼ 分類ルールを編集する

- 1 **Portal Server** 管理コンソールにログインします。
- 2 メニューバーから「検索サーバー」を選択し、検索サーバーを選択します。
- 3 「ロボット」を選択し、メニューバーから「分類ルール」を選択します。
- 4 分類ルールを選択して「編集」をクリックします。
- 5 必要に応じて属性を変更します。
属性の詳細については、「Manage Classification Rules」を『Sun Java System Portal Server 7.1 Technical Reference』で参照してください。
- 6 「了解」をクリックします。

ソースとデスティネーション

ほとんどのロボットアプリケーション関数 (RAF) は情報のソースを必要とし、デスティネーションに送信するデータを生成します。ソースはロボット内で定義され、最終的に生成されるリソース記述のフィールドに関連している必要はありません。一方、デスティネーションは通常、リソース記述サーバーのスキーマによって定義されるリソース記述のフィールド名になります。

次の節では、フィルタリングプロセスの各段階と、それらの段階で使用できるソースについて説明します。

- 162 ページの「セットアップ段階で使用可能なソース」
- 163 ページの「メタデータフィルタリング段階で使用可能なソース」
- 163 ページの「データ段階で使用可能なソース」
- 164 ページの「列挙、生成、およびシャットダウン段階で使用可能なソース」
- 164 ページの「プロパティーの有効化」

セットアップ段階で使用可能なソース

セットアップ段階では、フィルタが設定されますが、リソースの URL またはコンテンツについての情報は取得できません。

メタデータフィルタリング段階で使用可能なソース

メタデータ段階ではロボットはリソースの URL を検出しますが、リソースのコンテンツはダウンロードしていません。このため、`filter.conf` ファイルなどのほかのソースから派生するデータと、URL についての情報が使用可能です。ただし、この段階では、リソースのコンテンツについての情報は使用できません。

表 12-2 メタデータ段階で RAF で使用できるソース

ソース	説明	例
<code>csid</code>	カタログサーバー ID	<code>x-catalog//budgie.siroe.com:8086/alexandria</code>
<code>depth</code>	開始点からトラバースされたリンク数	10
<code>enumeration filter</code>	列挙フィルタの名前	<code>enumeration1</code>
<code>generation filter</code>	生成フィルタの名前	<code>generation1</code>
<code>host</code>	URL のホスト部分	<code>home.siroe.com</code>
<code>IP</code>	ホストの数値バージョン	198.95.249.6
<code>protocol</code>	URL のアクセス部分	<code>http, https, ftp, file</code>
<code>path</code>	URL のパス部分	<code>/, /index.html, /documents/listing.html</code>
<code>URL</code>	完全な URL	<code>http://developer.siroe.com/docs/manuals/</code>

データ段階で使用可能なソース

データ段階では、ロボットがその URL のリソースのコンテンツをダウンロードしているので、記述、作成者などのコンテンツに関するデータにアクセスできます。

リソースが HTML ファイルである場合、ロボットは HTML ヘッダーの `<META>` タグをパースします。したがって、`<META>` タグに含まれるデータは、データ段階で使用可能です。

データ段階では、メタデータ段階で使用可能なソースのほかに、次のソースを RAF で使用できます。

表 12-3 データ段階で RAF で使用できるソース

ソース	説明	例
content-charset	リソースが使用する文字セット	
content-encoding	エンコードの形式	
content-length	リソースのバイト単位のサイズ	
content-type	リソースの MIME タイプ	text/html、image/jpeg
expires	リソースが期限切れになる日付	
last-modified	リソースの最終更新日	
<META> タグ内のデータ	HTML リソースのヘッダーの <META> タグで提供されるデータ	作成者、記述、キーワード

<META> タグのデータを除くこれらのすべてのソースは、リソースの取得時に返される HTTP 応答ヘッダーから派生されます。

列挙、生成、およびシャットダウン段階で使用可能なソース

列挙および生成の段階では、データ段階と同じデータソースが使用可能です。詳細は、[表 12-3](#) を参照してください。

シャットダウン段階では、フィルタはフィルタリングを完了し、シャットダウンします。この段階用に書き込まれた関数は、データ段階で使用可能なソースと同じデータソースを使用できますが、シャットダウン関数の動作は通常、ロボットの終了およびクリーンアップに制限されます。

プロパティの有効化

各関数には enable プロパティがあります。値には、true、false、on、または off を指定できます。管理コンソールは、これらのパラメータを使用して特定のディレクトティブを有効または無効にします。

次の例は text/html の列挙を有効にし、text/plain の列挙を無効にします。

```
# Perform the enumeration on HTML only
Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
Enumerate enable=false fn=enumerate-urls-from-text max=1024 type=text/plain
```

`enable=false` プロパティまたは `enable=off` プロパティの追加は、行をコメントアウトするのと同じ効果があります。管理コンソールは、コメントを書き込まないため、これらのプロパティが使用されます。

セットアップ関数

ここでは、列挙および生成フィルタの両方によってセットアップ段階中に使用される関数について説明します。関数については、次の節で説明します。

filterrules-setup

`filterrules-setup` 関数を使用する場合、`logtype` ログファイルを使用します。値には、`verbose`、`normal`、または `terse` を指定できます。

プロパティ

`config` このフィルタが使用するフィルタールールを含むファイルのパス名。

例

```
Setup fn=filterrules-setup
```

```
config="/var/opt/SUNWportal/searchservers/search1/config/filterrules.conf"
```

setup-regex-cache

`setup-regex-cache` 関数は、`filter-by-regex` および `generate-by-regex` 関数のキャッシュサイズを初期化します。デフォルト値である 32 以外の数値を指定するときは、この関数を使用します。

プロパティ

`cache-size` `regex` キャッシュに保持されるコンパイル済み正規表現の最大数。

例

```
Setup fn=setup-regex-cache cache-size=28
```

setup-type-by-extension

setup-type-by-extension 関数はファイル名の拡張子を認識するようにフィルタを設定します。この関数を呼び出してから、assign-type-by-extension 関数を使用します。プロパティとして指定されたファイルは、標準の MIME コンテンツタイプとファイル拡張子の文字列の間のマッピングを含む必要があります。

プロパティ

ファイル MIME タイプ設定ファイルの名前。

例

```
Setup fn=setup-type-by-extension
```

```
file="/var/opt/SUNWportal/searchservers/search1/config/mime.types"
```

フィルタリング関数

フィルタリング関数はメタデータおよびデータの段階で動作し、関数およびそのプロパティによって指定された特定の条件に基づいて、リソースを許可または拒否します。これらの関数は、filter.conf ファイルの列挙フィルタおよび生成フィルタの両方で使用できます。

これらの各 filter-by 関数は比較を行い、リソースを許可または拒否します。リソースが許可されると、処理が次のフィルタリング段階へ進みます。リソースが拒否されると、リソースが列挙または生成を継続する条件を満たしていないという理由で処理が停止します。

filter-by-exact

filter-by-exact 関数は、allow/deny 文字列が情報のソースに完全に一致する場合、リソースを許可または拒否します。キーワード all は任意の文字列と一致します。

プロパティ

src 情報のソース。

allow/deny 文字列を含みます。

例

次の例は、content-type が text/plain のリソースすべてをフィルタして取り除きます。これにより、ほかのすべてのリソースの処理が続行されます。

```
Data fn=filter-by-exact src=type deny=text/plain
```

filter-by-max

filter-by-max 関数では、指定された情報ソースが指定値以下の場合にリソースが許可されます。情報ソースが指定された値よりも大きい場合は、リソースは拒否されます。

この関数は、1つのフィルタについて1回しか呼び出すことができません。

プロパティ

filter-by-max 関数は、filter-by-max 関数で使用するプロパティを一覧表示します。

src 情報のソース。hosts、objects、または depth

value 比較のための値を指定します。

例

この例は、content-length が 1024K バイトより小さいリソースを許可します。

```
MetaData fn=filter-by-max src=content-length value=1024
```

filter-by-md5

filter-by-md5 関数は、指定の MD5 チェックサム値を持つ最初のリソースだけを許可します。このロボットによって、現在のリソースの MD5 が以前のリソースで検出された場合、現在のリソースは拒否されます。この関数は、複数の URL を持つ同一のリソースまたは単一のリソースの重複を防ぎます。

この関数は、データ段階またはそれ以降の段階でだけ呼び出すことができます。この関数は、1つのフィルタについて1回しか呼び出すことができません。フィルタは generate-md5 関数を呼び出し、filter-by-md5 を呼び出す前に MD5 チェックサムを生成する必要があります。

プロパティ

なし

例

次の例は、MD5 チェックサムを処理する一般的な方法を示します。最初にチェックサムを生成し、そのチェックサムに基づいてフィルタリングします。

```
Data fn=generate-md5
```

```
Data fn=filter-by-md5
```

filter-by-prefix

`filter-by-prefix` 関数は、指定の情報ソースが指定されたプレフィックス文字列で開始する場合、リソースを許可または拒否します。リソースは、完全に一致する必要はありません。キーワード `all` は任意の文字列と一致します。

プロパティ

`src` 情報のソース。
`allow/deny` プレフィックス比較の文字列を含みます。

例

次の例は、`content-type` が `text/html` や `text/plain` など、任意の種類のテキストであるリソースを許可します。

```
MetaData fn=filter-by-prefix src=type allow=text
```

filter-by-regex

`filter-by-regex` 関数は、正規表現によるパターンマッチングをサポートします。この関数は、指定の正規表現に一致するリソースを許可します。サポートされる正規表現構文は、POSIX.1 仕様によって定義されます。* の正規表現は、任意の文字に一致します。

プロパティ

`src` 情報のソース。
`allow/deny` 正規表現の文字列を含みます。

例

次の例は、政府(.gov)ドメインのサイトからのすべてのリソースを拒否します。

```
MetaData fn=filter-by-regex src=host deny=\\.gov
```

filterrules-process

filterrules-process 関数は、filterrules.conf ファイルのサイト定義とフィルタルールを処理します。

プロパティ

なし

例

```
MetaData fn=filterrules-process
```

サポート関数のフィルタリング

サポート関数は、リソース上で情報を操作または生成するため、フィルタリング中に使用されます。ロボットは、フィルタリング関数を呼び出すことによって、リソースを処理することができます。これらの関数は、ファイル filter.conf の列挙および生成のフィルタで使用することができます。

assign-source

assign-source 関数は、指定した情報ソースに新規の値を割り当てます。この関数により、フィルタリング処理中の編集が可能になります。この関数は、明示的な新規値を割り当てるか、または、別の情報ソースから値をコピーすることができます。

プロパティ

dst 値を変更するソースの名前。

value 明示的な値を指定。

src dst にコピーする情報ソース。

value プロパティまたは src プロパティのどちらかを指定する必要がありますが、両方は指定しません。

例

```
Data fn=assign-source dst=type src=content-type
```

assign-type-by-extension

`assign-type-by-extension` 関数は、リソースのファイル名を使用してタイプを決定し、処理を継続するリソースにこのタイプを割り当てます。

`assign-type-by-extension` を使用可能にするには、セットアップ中に `setup-type-by-extension` 関数を呼び出す必要があります。

プロパティ

`src` 比較するファイル名のソース。ソースを指定しない場合、デフォルトではリソースのパスになります。

例

```
MetaData fn=assign-type-by-extension
```

clear-source

`clear-source` 関数は、指定されたデータソースを削除します。通常は、この関数を実行する必要はありません。 `assign-source` 関数を使用することにより、ソースを作成または置換することができます。

プロパティ

`src` 削除するソースの名前。

例

次の例は、パスのソースを削除します。

```
MetaData fn=clear-source src=path
```

convert-to-html

`convert-to-html` 関数は、リソースのタイプが指定の MIME タイプと一致する場合、現在のリソースを HTML ファイルに変換して処理を継続します。変換フィルタは、変換中のファイルのタイプを自動的に検出します。

プロパティ

type 変換を行う MIME タイプ。

例

次の関数呼び出しのシーケンスにより、フィルタは、すべての Adobe Acrobat PDF ファイル、Microsoft RTF ファイル、および FrameMaker MIF ファイル、およびファイルを配信するサーバーによってタイプが指定されていないファイルを HTML に変換します。

```
Data fn=convert-to-html type=application/pdf
```

```
Data fn=convert-to-html type=application/rtf
```

```
Data fn=convert-to-html type=application/x-mif
```

```
Data fn=convert-to-html type=unknown
```

copy-attribute

copy-attribute 関数は、リソース記述の 1 つのフィールドからの値を別のフィールドにコピーします。

プロパティ

src コピー元のリソース記述のフィールド。

dst ソースのコピー先のリソース記述の項目。

truncate コピーするソースの最大長。

clean 省略されたテキストを修正するかどうか、分割された単語を修正するかを示す論理値プロパティ。デフォルトでは、このプロパティは `false` に設定されています。

例

```
Generate fn=copy-attribute \\  
src=partial-text dst=description truncate=200 clean=true
```

generate-by-exact

generate-by-exact 関数は、既存のソースが別の値と完全に一致する場合にかぎり、指定された値を持つソースを生成します。

プロパティ

dst 生成するソースの名前。
value dst に割り当てる値。
src 比較対象のソース。

例

次の例は、ホストが `www.siroe.com` の場合に、分類を `siroe` に設定します。

```
Generate fn="generate-by-exact" match="www.siroe.com:80" src="host"  
value="Siroe" dst="classification"
```

generate-by-prefix

この generate-by-prefix 関数は、既存のソースのプレフィックスが別の値と一致する場合に、指定された値を持つソースを生成します。

プロパティ

dst 生成するソースの名前。
value dst に割り当てる値。
src 比較対象のソース。
match src と比較する値。

例

次の例は、プロトコルプレフィックスが `HTTP` の場合、分類を `World Wide Web` に設定します。

```
Generate fn="generate-by-prefix" match="http" src="protocol" value="World Wide  
Web" dst="classification"
```

generate-by-regex

generate-by-regex 関数は、既存のソースが正規表現に一致する場合に、指定した値を持つソースを生成します。

プロパティ

dst 生成するソースの名前。
value dst に割り当てる値。
src 比較対象のソース。
match src と比較する正規表現の文字列。

例

次の例は、ホスト名が正規表現 `*.siroe.com` に一致する場合に、分類を `siroe` に設定します。たとえば、`developer.siroe.com` および `home.siroe.com` の両方にあるリソースは、`siroe` として分類されます。

```
Generate fn="generate-by-regex" match="\\\\"*\\.siroe\\.com" src="host"
value="Siroe" dst="classification"
```

generate-md5

generate-md5 関数は、MD5 チェックサムを生成し、リソースに追加します。これにより、filter-by-md5 関数を使用して、重複する MD5 チェックサムを持つリソースを拒否することができます。

プロパティ

なし

例

```
Data fn=generate-md5
```

generate-rd-expire

generate-rd-expire 関数は、有効期限を生成して、指定されたソースに追加します。この関数は、HTTP ヘッダーおよび HTML `<META>` タグなどのメタデータを使用し、リソースから任意の有効期限を取得します。有効期限が存在しない場合は、現在の日付から 3 ヶ月間の有効期限を生成します。

プロパティ

`dst` ソースの名前。指定を省略した場合、ソースのデフォルトは `rd-expires` となります。

例

Generate `fn=generate-rd-expires`

generate-rd-last-modified

`generate-rd-last-modified` 関数は、現在の時間を指定されたソースに追加します。

プロパティ

`dst` ソースの名前。指定を省略した場合、ソースのデフォルトは `rd-last-modified` となります。

例

Generate `fn=generate-last-modified`

rename-attribute

`rename-attribute` 関数は、リソース記述のフィールドの名前を変更します。この関数は、たとえば、`extract-html-meta` で `<META>` タグからフィールドに情報をコピーし、そのフィールドの名前を変更する場合にとっても便利です。

プロパティ

`src` 1つの名前から別の名前へのマッピングを含む文字列。

例

次の例は、属性の名前を `author` から `author-name` に変更します。

Generate `fn=rename-attribute src="author->author-name"`

列挙関数

次の関数は、列挙段階で動作します。これらの関数は、ロボットが指定のリソースからリンクを収集するかどうか、およびその方法を制御し、リソースの発見を継続する開始点として使用します。

enumerate-urls

enumerate-urls 関数はリソースをスキャンし、ハイパーテキストリンクで検出されるすべての URL を列挙します。結果を使用して、リソース発見を継続します。コンテンツタイプを指定し、列挙される URL の種類を制限することができます。

プロパティ

max 指定のリソースから生成する URL の最大数。デフォルトは 1024 となります。

type 指定したコンテンツタイプの URL に列挙を制限するコンテンツタイプ。type はオプションのプロパティです。指定を省略した場合、すべての URL が列挙されます。

例

次の例は、HTML URL だけを、最大値 1024 まで列挙します。

```
Enumerate fn=enumerate-urls type=text/html
```

enumerate-urls-from-text

enumerate-urls-from-text 関数はテキストリソースをスキャンし、次の正規表現に一致する文字列を検索します。URL:.*. この関数はロボットを生成し、これらの文字列から URL を列挙し、さらにリソース記述を生成します。

プロパティ

max 指定のリソースから生成する URL の最大数。指定を省略した場合、デフォルトは 1024 となります。

例

```
Enumerate fn=enumerate-urls-from-text
```

生成関数

生成関数は、フィルタリングの生成段階で使用されます。生成関数は、リソース記述に入る情報を作成することができます。通常、リソースの本体から情報を抽出するか、リソースのメタデータから情報をコピーします。

extract-full-text

extract-full-text 関数は、リソースの完全なテキストを抽出して、リソース記述に追加します。

注 - extract-full-text 関数の使用には注意が必要です。リソース記述のサイズを非常に大きくすることができるため、データベースが膨大化したり、ネットワーク帯域幅全体に悪影響を及ぼしたりする可能性があります。

例

```
Generate fn=extract-full-text
```

プロパティ

truncate リソースから抽出する文字の最大数。

dst 完全なテキストを受信するスキーマ項目の名前。

extract-html-meta

extract-html-meta 関数は、HTML ファイルから任意の <META> または <TITLE> 情報を抽出し、リソース記述に追加します。コンテンツタイプの指定により、生成する URL の種類が制限される場合があります。

プロパティ

truncate 抽出する最大バイト数。

type 省略可能なプロパティ。指定を省略した場合は、すべての URL が生成されます。

例

```
Generate fn=extract-html-meta truncate=255 type=text/html
```

extract-html-text

extract-html-text 関数は、HTML タグを除いた、テキストの最初の数文字を HTML ファイルから抽出し、そのテキストをリソース記述に追加します。この関数により、ドキュメントのテキストの最初の部分を RD に含むことができます。コンテンツタイプの指定により、生成する URL の種類が制限される場合があります。

プロパティ

truncate	抽出する最大バイト数。
skip-headings	true に設定すると、ドキュメント内で発生する任意の HTML ヘッダーは無視されます。
type	省略可能なプロパティ。指定を省略した場合は、すべての URL が生成されます。

例

```
Generate fn=extract-html-text truncate=255 type=text/html skip-headings=true
```

extract-html-toc

extract-html-toc 関数は、HTML ヘッダーから目次を抽出し、リソース記述に追加します。

プロパティ

truncate	抽出する最大バイト数。
level	抽出する HTML ヘッダーの最大レベル。このプロパティは、目次の深さを制御します。

例

```
Generate fn=extract-html-toc truncate=255 level=3
```

extract-source

extract-source 関数は、指定のソースから特定の値を抽出し、リソース記述に追加します。

プロパティ

src ソース名のリスト。->演算子を使用し、RD 属性の新規名を定義できます。
たとえば、type->content-type は type という名前のソースの値を取得し、
content-type という名前の属性で RD に保存されます。

例

```
Generate fn=extract-source src="md5,depth,rd-expires,rd-last-modified"
```

harvest-summarizer

harvest-summarizer 関数は、リソース上で Harvest サマライザを実行し、結果をリソース記述に追加します。

Harvest サマライザを実行するには、ロボットを実行する前に、path に \$HARVEST_HOME/lib/gatherer を設定する必要があります。

プロパティ

summarizer サマライザプログラムの名前。

例

```
Generate fn=harvest-summarizer summarizer=HTML.sum
```

シャットダウン関数

filterrules-shutdown 関数は、列挙および生成の両関数によってシャットダウン段階中に使用できます。

filterrules-shutdown

ルールが実行されたあと、filterrules-shutdown 関数はクリーンアップおよびシャットダウンを実行します。

プロパティ

なし

例

```
Shutdown fn=filterrules-shutdown
```

変更可能なプロパティ

robot.conf ファイルは、filter.conf にある適切なフィルタにロボットを指定するなど、ロボット用の多くのオプションを定義します。旧バージョンとの下位互換性のために、robot.conf に開始ポイント URL を含めることもできます。

管理コンソールを使用すれば、ほとんどのプロパティを設定できるので、通常は robot.conf ファイルを編集する必要はありません。ただし、上級ユーザーはこのファイルを手動で編集することにより、管理コンソールを介してでは設定できないプロパティを設定することができます。このファイルの例については、[186 ページの「robot.conf ファイルの例」](#)を参照してください。

表 12-4 は、robot.conf ファイルでユーザーが変更可能なプロパティを示しています。

表 12-4 ユーザーが変更できるプロパティ

プロパティ	説明	例
auto-proxy	ロボットのプロキシ設定を指定します。これは、プロキシサーバー、またはプロキシを自動的に設定する Java スクリプトファイルです。	auto-proxy="http://proxy_server/proxy.pac"
bindir	ロボットが bin ディレクトリを PATH 環境に追加するかどうかを指定します。これは cmd-hook プロパティが指定するプログラムなど、ユーザーがロボットで外部プログラムを実行するための特別な PATH です。	bindir=path

表 12-4 ユーザーが変更できるプロパティー (続き)

プロパティー	説明	例
cmd-hook	<p>ロボットが実行を完了するたびに実行される外部完了スクリプトを指定します。これは、コマンド名の完全パスで指定する必要があります。ロボットは /var/opt/SUNWportal/ ディレクトリからこのスクリプトを実行します。</p> <p>デフォルトはありません。</p> <p>コマンドを実行するには、最低1つの RD が登録されている必要があります。</p>	cmd-hook=" command-string "
command-port	<p>管理インタフェースまたはロボットコントロールパネルのような、ほかのプログラムからコマンドを受け入れるために、ロボットが待機するポート番号を指定します。</p> <p>セキュリティ上の理由から、remote-access が yes に設定されていない場合、ロボットはローカルホストからのコマンドだけを受け入れます。</p>	command-port=port_number
connect-timeout	<p>ネットワークが接続要求に回答する最大許容時間を指定します。</p> <p>デフォルトは 120 秒です。</p>	command-timeout=seconds
convert-timeout	<p>ドキュメント変換に許可される最大時間を指定します。</p> <p>デフォルトは 600 秒です。</p>	convert-timeout=seconds

表 12-4 ユーザーが変更できるプロパティ (続き)

プロパティ	説明	例
depth	<p>ロボットが調べる開始ポイント URL からのリンク数を指定します。このプロパティは、深さを指定しない開始ポイント URL のデフォルト値を設定します。</p> <p>デフォルトは 10 です。</p> <p>マイナス 1 の値 (depth=-1) は、リンクの深さが無限であることを表します。</p>	depth=integer
email	<p>ロボットを実行するユーザーの電子メールアドレスを指定します。</p> <p>Web 管理者とサイトでロボットを実行するユーザーとが連絡できるように、電子メールアドレスは HTTP 要求ヘッダーで user-agent とともに送信されます。</p> <p>デフォルトは user@domain です。</p>	email=user@hostname
enable-ip	<p>作成する各 RD の URL に IP アドレスを生成します。</p> <p>デフォルトは true です。</p>	enable-ip=[true yes false no]
enable-rdm-probe	<p>サーバーが RDM をサポートするかどうかを確認します。ロボットはこのプロパティを使用して、検出する各サーバーに照会するかどうかを決定します。サーバーが RDM をサポートする場合は、サーバーは独自のリソース記述サーバーの役割を果たすことができるため、ロボットはこのサーバーのリソースを列挙しません。</p> <p>デフォルトは false です。</p>	enable-rdm-probe=[true false yes no]

表 12-4 ユーザーが変更できるプロパティ (続き)

プロパティ	説明	例
enable-robots-txt	<p>ロボットがアクセスする各サイトで、robots.txt ファイルが使用できる場合、このファイルをロボットが検査する必要があるかどうかを指定します。</p> <p>デフォルトは yes です。</p>	enable-robots-txt=[true false yes no]
engine-concurrent	<p>ロボットが使用する事前作成のスレッド数を指定します。</p> <p>デフォルトは 10 です。</p> <p>管理コンソールを使用して、このプロパティを対話形式で設定することはできません。</p>	engine-concurrent=[1..100]
enumeration-filter	<p>リソースを列挙すべきかどうかを確認する場合に、ロボットが使用する列挙フィルタを指定します。値は filter.conf ファイルで定義されるフィルタ名にする必要があります。</p> <p>デフォルトは enumeration-default です。</p> <p>管理コンソールを使用して、このプロパティを対話形式で設定することはできません。</p>	enumeration-filter=enumfiltername
generation-filter	<p>リソース用にリソース記述を生成する必要があるかどうかを確認する場合に、ロボットが使用する生成フィルタを指定します。値は filter.conf ファイルで定義されるフィルタ名にする必要があります。</p> <p>デフォルトは generation-default です。</p> <p>管理コンソールを使用して、このプロパティを対話形式で設定することはできません。</p>	generation-filter=genfiltername

表 12-4 ユーザーが変更できるプロパティ (続き)

プロパティ	説明	例
index-after-ngenerated	<p>検索サーバー用にバッチ処理する前に、ロボットが RD を収集する時間を分単位で指定します。</p> <p>デフォルト値は、30 分です。</p>	index-after-ngenerated=30
logLevel	<p>ロギングのレベルを指定します。logLevel の値は次のように指定されます。</p> <ul style="list-style-type: none"> ■ レベル 0: 重大なエラー以外は記録しません ■ レベル 1: RD 生成も記録します (デフォルト) ■ レベル 2: 検索状況も記録します ■ レベル 3: フィルタリング状況も記録します ■ レベル 4: 生成状況も記録します ■ レベル 5: 検索の進行状況も記録します <p>デフォルト値は 1 です。</p>	logLevel=[0...100]
max-connections	<p>ロボットが実行できる並行検索の最大数を指定します。</p> <p>デフォルトは 8 です。</p>	max-connections=[1..100]
max-filesize-kb	<p>ロボットが検索するファイルの最大ファイルサイズを K バイト単位で指定します。</p>	max-filesize-kb=1024
max-memory-per-url / max-memory	<p>各 URL が使用する最大メモリーをバイト単位で指定します。URL がより多くのメモリーを必要とする場合、RD はディスクに保存されます。</p> <p>デフォルトは 64k です。</p> <p>管理コンソールを使用して、このプロパティを対話形式で設定することはできません。</p>	max-memory-per-url=n_bytes

表 12-4 ユーザーが変更できるプロパティ (続き)

プロパティ	説明	例
max-working	<p>ロボット稼働セットのサイズ、すなわちロボットが一度に処理できる URL の最大数を指定します。</p> <p>管理コンソールを使用して、このプロパティを対話形式で設定することはできません。</p>	max-working=1024
onCompletion	<p>実行完了後のロボットの動作を決定します。ロボットはアイドルモードになるか、ループバック、再起動、または終了することができます。</p> <p>デフォルトは idle です。</p> <p>このプロパティは cmd-hook プロパティとともに動作します。実行後のロボットは、onCompletion アクションの次に、cmd-hook プログラムを実行します。</p>	OnCompletion=[idle loop quit]
password	<p>httpd 認証および ftp 接続に使用される password を指定します。</p>	password=string
referer	<p>Web ページにアクセスする際にロボットを Referer として識別するように設定されている場合、HTTP 要求に送信されるプロパティを指定します。</p>	referer=string
register-user	<p>検索サーバー データベースに RD を登録するために使用されるユーザー名を指定します。</p> <p>このプロパティは、検索サーバー 管理インタフェースから対話形式で設定することはできません。</p>	register-user=string
register-password	<p>検索サーバー データベースに RD を登録するために使用されるパスワードを指定します。</p> <p>このプロパティは、管理コンソールから対話形式で設定することはできません。</p>	register-password=string

表 12-4 ユーザーが変更できるプロパティ (続き)

プロパティ	説明	例
remote-access	ロボットが、リモートホストからのコマンドを受け入れることができるかどうかを指定します。 デフォルトは false です。	remote-access=[true false yes no]
robot-state-dir	ロボットが状態を保存するディレクトリを指定します。この作業ディレクトリで、ロボットは収集された RD の数などを記録できます。	robot-state-dir="/var/opt/SUNWportal/searchservers/<searchserverid>/config/robot"
server-delay	同じ Web サイトに次にアクセスするまでの時間間隔を指定します。この時間を指定することで、ロボットが頻繁に同じサイトにアクセスするのを防止できます。デフォルトは 0 秒です。	server-delay=delay_in_seconds
site-max-connections	ロボットが任意の 1 つのサイトに対して行う同時接続の最大数を示します。 デフォルトは 2 です。	site-max-connections=[1..100]
smart-host-heuristics	ロボットが正規の DNS ホスト名を巡回するサイトを変更できます。たとえば、 www123.siroe.com が www.siroe.com に変更されます。 デフォルトは false です。	smart-host-heuristics=[true false]
tmpdir	ロボットが一時ファイルを作成する場所を指定します。 環境変数 TMPDIR を設定する際に、この変数を使用します。	tmpdir=path
user-agent	http-request 内の電子メールアドレスとともにサーバーに送信されるプロパティを指定します。	user-agent=SunONERobot/6.2
username	ロボットを実行するユーザーのユーザー名を指定します。これは、http 認証および ftp 接続で使用されます。 デフォルトは anonymous です。	username=string

robot.confファイルの例

ここでは、robot.conf ファイルの例について説明します。この例でコメントが付いたプロパティーには、表示されているデフォルト値が適用されます。最初のプロパティーである csid は、このファイルを使用する 検索サーバー インスタンスを示しています。このプロパティーの値は変更しないでください。このファイルのプロパティーの定義については、[179 ページの「変更可能なプロパティー」](#)を参照してください。

注- このサンプルファイルには、検索サーバーによって使用される、変更できないプロパティーがいくつか含まれています。csid プロパティーは、その一つです。

```
<Process csid="x-catalog://budgie.siroe.com:80/jack" \<\  
  auto-proxy="http://sesta.varrius.com:80/"  
  auto_serv="http://sesta.varrius.com:80/"  
  command-port=21445  
  convert-timeout=600  
  depth="-1"  
  # email="user@domain"  
  enable-ip=true  
  enumeration-filter="enumeration-default"  
  generation-filter="generation-default"  
  index-after-ngenerated=30  
  loglevel=2  
  max-concurrent=8  
  site-max-concurrent=2  
  onCompletion=idle  
  password=boots  
  proxy-loc=server  
  proxy-type=auto  
  robot-state-dir="/var/opt/SUNWportal/searchservers/search1/robot" \<\  
  ps/robot"  
  server-delay=1  
  smart-host-heuristics=true  
  tmpdir="/var/opt/SUNWportal/searchservers/search1/tmp"  
  user-agent="iPlanetRobot/4.0"  
  username=jack  
</Process>
```

パート III

委任管理の管理

- 第13章
- 第14章

委任管理チャネルの管理

Portal Server では、ポータル管理者が、特定の組織内の各種のタスクを管理する責任を委任管理者と呼ばれるほかのユーザーに委任することができます。管理機能を分散させると、とりわけ複雑な組織でポータルの管理を改善できます。ポータルの管理者は、委任管理者がデスクトップの管理に使用するチャネルを設定できます。

委任管理者は、Portal Server デスクトップにある管理ポートレットを使用して、管理タスクを実行します。ここでは、Developers Sample デスクトップにこれらのチャネルを設定する方法について説明します。これにより、委任管理者用の基本的なデスクトップを設計できるようになります。

- 189 ページの「ポータルの委任管理の概要」
- 190 ページの「委任管理チャネルの設定」

ポータルの委任管理の概要

Portal Server では、Portal Server デスクトップで管理ポートレットを提供しています。管理者は、これらのポートレットを使用して、委任管理者がデスクトップとエンドユーザーロールの管理に使用する特殊なチャネルを設定できます。次の3つの委任管理ロールがあります。

- 組織管理者ロール — 定義された組織内のデスクトップコンテンツおよびエンドユーザーを管理します。
- コンテンツ管理ロール — 定義された組織内のエンドユーザーのデスクトップコンテンツを管理します。
- ユーザー管理ロール — 定義された組織内のエンドユーザーを管理し、エンドユーザーロールを割り当てたり、割り当てを解除することができます。

ここでは、Developers Sample デスクトップにこれらのチャネルを設定する方法について説明します。これにより、委任管理者用の基本的なデスクトップを設計できるようになります。

委任管理チャネルの設定

ここでは、Developers Sample デスクトップで、委任管理チャネルを組織、ロール、およびユーザーのレベルに設定する方法を示します。

▼ 委任管理チャネルを設定する

- 1 デスクトップチャネルへのアクセスを許可または制限するアクセス制御命令を設定します。
 - 組織レベルの管理者アクセスの場合、アクセス制御命令は **Access Manager** によってデフォルトで設定されます。
 - ロールレベルまたはユーザーレベルの管理者アクセスの場合は、**Portal Server** の管理者がアクセス制御命令を設定する必要があります。

- a. サンプル ACI (アクセス制御命令) を **Directory Server** にロードします。

```
ldapmodify -D "cn=directory manager" -w -f acis.ldif と入力します。
```

サンプル ACI の内容は次のとおりです。

```
#
acis.ldif
dn:dc=sample,dc=siroe,dc=com
changetype:modify

# JDCAdmin1 ロールの ACI

add:aci
aci: (target= "ldap:///ou=people,o=DeveloperSample,dc=red,dc=iplanet,dc=com")
(targetattr = "**")
(version 3.0; acl "Allow JDCAdmin1 Role to read and search users";
allow (read,search)
roledn = "ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com");
-
add:aci
aci: (target="ldap:///dc=red,dc=iplanet,dc=com")
(targetfilter="(entrydn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com)")
(targetattr="**")
(version 3.0; acl "Allow JDCAdmin1 Role to read and search JDC Role";
allow (read,search)
roledn="ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com");
-
add:aci
```

```

aci: (target="ldap:///ou=people,o=DeveloperSample,dc=red,dc=iplanet,dc=com")
(targetattr="nsroledn")
(targetfilter="(!(|(nsroledn=cn=Top-level Admin Role,dc=red,dc=iplanet,dc=com)
(nsroledn=cn=Top-level Help Desk Admin Role,dc=red,dc=iplanet,dc=com)
(nsroledn=
cn=Organization Admin Role,o=DeveloperSample,dc=red,dc=iplanet,dc=com)
(nsroledn=cn=Top-level Policy Admin Role,dc=red,dc=iplanet,dc=com))))")
(targetattrfilters="add=nsroledn:
(nsroledn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com),
del=nsroledn:(nsroledn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com)")
(version 3.0; acl "Allow JDCAdmin1 Role to add/remove users to JDC Role";
allow (write)
roledn="ldap:///cn=JDCAdmin1,o=DeveloperSample,dc=red,dc=iplanet,dc=com");)
-

# JDCAdmin2 ロールの ACI

add:aci
aci:
(target="ldap:///cn=SunPortalportal1DesktopService,dc=red,dc=iplanet,dc=com")
(targetfilter=
(cn=cn=JDC,o=DeveloperSample,dc=red,dc=iplanet,dc=com))(targetattr="*")
(version 3.0; acl "Allow JDCAdmin2 to edit display profile of JDC Role";
allow (all)
roledn="ldap:///cn=JDCAdmin2,o=DeveloperSample,dc=red,dc=iplanet,dc=com");)
-
add:aci
aci: (target="ldap:///dc=red,dc=iplanet,dc=com")(targetattr = "*")
(version 3.0; acl "Allow JDCAdmin2 to read and search all";
allow (read,search)
roledn = "ldap:///cn=JDCAdmin2,o=DeveloperSample,dc=red,dc=iplanet,dc=com");)

#

```

- b. `o=DeveloperSample,dc=red,dc=iplanet` となっている部分をすべて `dc=sample,dc=hostname,dc=com` に置き換えます。

2 委任管理者のロールを定義します。

a. Sun Java™ System Access Manager 管理コンソールにログインします。

Access Manager コンソールについては、『Sun Java System Access Manager 7.1 管理ガイド』を参照してください。

b. DeveloperSample 組織に移動します。

c. 次のいずれかを作成します。

■ 新しいサブ組織

新しい組織を作成すると、Access Manager によってその組織の組織管理者ロールが設定されます。

i. この新しい組織に必要なすべてのサービスを登録します。

ii. 新しいユーザーを作成し、このユーザーに組織管理者ロールを割り当てます。

■ 新しい委任管理ロール

i. 次の新しいロールを作成します。

■ エンドユーザーロール — ロール JDC を作成し、「タイプ」を「サービス」に設定して、アクセス権をオフにします。

■ コンテンツ管理ロール — ロール JDCAdmin2 を作成し、「タイプ」を「管理」に設定して、アクセス権をオフにします。

■ ユーザー管理ロール — ロール JDCAdmin1 を作成し、「タイプ」を「ユーザー」に設定して、アクセス権をオフにします。

ii. 次の新しいユーザーを作成します。

■ jdcuser — ロール JDC に割り当てます。

■ jdcuadmin — ロール JDCAdmin1 に割り当てます。

■ jdcadmin — ロール JDCAdmin2 に割り当てます。

d. (省略可能) Access Manager コンソールからログアウトします。

3 ポータルデスクトップサービスの属性値が、管理者ロール DN 側とポータル側で一致するようにします。

管理者ロール DN のデスクトップサービス属性値は、次のとおりです。

■ content.admin.role.dn

■ user.admin.role.dn

ポータルのデスクトップサービスの属性値がこれらの値と一致しない場合、この管理者ロールに属するユーザーがポータルに対して認証を行うと、正しくないデスクトップが表示されることがあります。

たとえば、DeveloperSample ポータルデスクトップサービスの属性値を次のように設定したとします。

■ 親コンテナ: JSPTabContainer

■ EditContainer: JSPEditContainer

- デフォルトタイプ: `developer_sample`

一方、管理者ロール DN を両方とも次のように設定したとします。

```
cn=Organization Admin Role, o=DeveloperSample, dc=siroe, dc=com
```

この場合、管理者ロール DN に対応するポータルデスクトップサービスの属性を次のように設定する必要があります。

```
cn=Organization Admin Role, o=DeveloperSample, dc=siroe, dc=com
```

- 4 taskadmin.properties ファイルを編集します。
 - a. `portal-base-directory/samples/taskadmin` ディレクトリの `taskadmin.properties` ファイルを開きます。
 - b. 次の変数の値を指定します。
 - `am.admin.dn` — 最上位管理者 DN (たとえば、`amadmin`)
 - `default.org.dn` — 最上位組織またはデフォルト組織 (たとえば、`dc=sun,dc=com`)
 - `ps.portal.id` — Portal ID (たとえば、`portal1`)
 - `ps.parent.tab.container` — ポータルデスクトップの親コンテナ名 (たとえば、`ASCTabContainer`)
 - `ps.default.type` — ポータルデスクトップタイプ (たとえば、`enterprise_sample`)
 - `content.admin.role.dn` — コンテンツ管理者のチャネルとコンテナがロードされる DN
 - `user.admin.role.dn` — ユーザー管理者のチャネルとコンテナがロードされる DN
 - `managed.content.dn` — コンテンツ管理者ロールで管理される DN
 - c. 実際の配備に合わせてデフォルト値を変更します。

```
# -----
# 全般設定
# -----
#
# psadmin パスワードファイル (ファイル名とディレクトリパス)
# 例: /tmp/password
#
psadmin.password.file=/tmp/password //パスワードファイルにパスワードが含まれます
#
# ポータル構成場所
# 例: /etc/opt/SUNWportal
#
ps.config.location=/etc/opt/SUNWportal
#
```

```
# Portal ID
# 例: portal1
#
ps.portal.id=portal1
#
# Access Manager 管理者 DN
# 例: uid=amAdmin,ou=People,dc=siroe,dc=com
#
am.admin.dn=uid=amAdmin,ou=People,dc=siroe,dc=com
#
# Access Manager デフォルト組織
# 例: dc=siroe,dc=com
#
default.org.dn=dc=siroe,dc=com
#
# -----
# タスク管理全般設定
# -----
#
# 親タブコンテナ
# 例: JSPTabContainer
#
ps.parent.tab.container=JSPTabContainer
#
# 親タブコンテナプロバイダ
# 例: JSPTabContainerProvider
#
ps.parent.tab.container.provider=JSPTabContainerProvider
#
# ポータルデフォルトタイプ
# 例: developer_sample
#
ps.default.type=developer_sample
#
# -----
# コンテンツ管理者設定
# -----
#
# コンテンツ管理者ロール DN。コンテンツ管理チャネルおよびコンテナ
# がこの DN にロードされます。# 例: 次を参照してください
#
content.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# 管理対象コンテンツ DN。'content.admin.role.dn' で管理される DN です。
```

```

# 例: 次を参照してください
#
managed.content.dn=o=DeveloperSample,dc=siroe,dc=com
#
# -----
# ユーザー管理者設定
# -----
#
# ユーザー管理者ロール DN。ユーザー管理チャネルおよびコンテナ
# がこの DN にロードされます。# 例: 次を参照してください
#
user.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# =====
# 例
# =====
#
# 組織管理者の例:
# content.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
# managed.content.dn=o=DeveloperSample,dc=siroe,dc=com
# user.admin.role.dn=cn=Organization Admin Role,o=DeveloperSample,dc=siroe,dc=com
#
# ロール管理者の例:
# content.admin.role.dn=cn=JDCAdmin2,o=DeveloperSample,dc=siroe,dc=com
# managed.content.dn=cn=JDC,o=DeveloperSample,dc=siroe,dc=com
# user.admin.role.dn=cn=JDCAdmin1,o=DeveloperSample,dc=siroe,dc=com

```

d. ant コマンドを実行します。

```

/usr/sfw/bin/ant -f ps-base-directory/samples/taskadmin/build.xml
-Dprops.location=/tmp

```

tmp は *taskadmin.properties* ファイルの場所です

5 追加されたことを確認します。

a. 新しい委任管理者のユーザーデスクトップにログインします。

b. 新しい委任管理チャネルを表示します。

- 組織委任管理者の場合は、この組織の管理チャネルが **Developer Sample** の「管理」タブに表示されることを確認します。
- ロールまたはユーザー委任管理者の場合は、このユーザーの管理チャネルが **Developer Sample** の「管理」タブに表示されることを確認します。

- c. ユーザーデスクトップからログアウトします。

Portal Server 委任管理タグライブラリの使用

Portal Server 委任管理タグライブラリを使用すると、次の操作が可能です。

- 事前に設定された委任管理ポートレットの変更
- 新しい委任管理機能を備えたポートレットの開発
- カスタムユーザーインタフェースを備えた管理ポートレットの記述
- JSPProvider に基づくチャンネルの作成と管理

委任管理タグライブラリの概要

『Tag Library for Delegated Administration』で、委任管理ポートレットを記述するためのタグおよびそれらの構文について説明しています。タグライブラリでサポートする管理機能タスクは次のとおりです。

- プロバイダ管理
- ポートレット管理
- ユーザー管理
- WSRP 管理

▼ 委任管理タグのリファレンスを使用する

『Tag Library for Delegated Administration』にタグ名と構文が記載されています。

- 1 『Tag Library for Delegated Administration』に移動します。
- 2 表示する内容を選択します。
 - タイトルを展開して、選択可能なセクションを表示します。
 - Tags for Desktop Channel and Container Management Tasks
 - Tags for Portlet Management Tasks
 - Tags for User Management Tasks

- Tags for WSRP (Web Services for Remote Portlets) Management Tasks
- タイトルリンクをクリックすると、リファレンスが先頭から表示されます。

索引

A

Access Manager

Portal Server, 44

コンソール, 44

assign-source 関数, 169-170

assign-type-by-extension 関数, 170

C

clear-source 関数, 170

convert-to-html 関数, 170-171

copy-attribute 属性, 171

D

DiscussionLite チャンネル, 125

DiscussionProvider

概要, 122-123

削除, 124-125

作成, 123-125

設定, 125

E

enumerate-urls-from-text 関数, 175

enumerate-urls 関数, 175

extract-full-text 関数, 176

extract-html-meta 関数, 176

extract-html-text 関数, 177

extract-html-toc 関数, 177

extract-source 関数, 177-178

F

filter-by-exact 関数, 166-167

filter-by-max 関数, 167

filter-by-md5 関数, 167-168

filter-by-prefix 関数, 168

filter-by-regex 関数, 168-169

filterrules-process 関数, 169

filterrules-setup 関数, 165

filterrules-shutdown 関数, 178-179

G

generate-by-exact 関数, 172

generate-by-prefix 関数, 172

generate-by-regex 関数, 173

generate-md5 関数, 173

generate-rd-expires 関数, 173-174

generate-rd-last-modified 関数, 174

H

harvest-summarizer 関数, 178

L

LDAP ディレクトリノード, 49-51

P

Portal Server

- インスタンス, 39
- 監視, 103
- 管理コンソールの使用, 29
- コマンド行インタフェースの使用, 31
- コンポーネント, 27-28

Portal Server インスタンス

- 一覧の表示, 39
- 概要, 39
- 表示, 39

psadmin コマンド

- Web Server との関係, 31
- コマンド行インタフェース, 31

R

RD, 「リソース記述」を参照

rename-attribute 関数, 174

S

setup-regex-cache 関数, 165

setup-type-by-extension 関数, 166

W

Web Server, psadmin コマンドとの関係, 31

WSRP コンシューマ

- 設定済みプロデューサの追加, 85-86
- 属性のマッピング, 92
- 追加, 97
- 名前の指定, 98
- プロキシの設定, 92-93
- 変更, 97-98

WSRP 標準, 概要, 75-76

WSRP プロデューサ

- WSRP コンシューマの登録, 95-96
- 管理, 76-85
- 検索, 83-85
- 作成, 93-94
- ダイジェストパスワード, 88-89

WSRP プロデューサ (続き)

- 登録ハンドル, 80-81
- プロパティ, 79
- 変更, 94-95
- レジストリサーバーと, 81-83

い

委任管理

- 概要, 30
- 管理, 189-196
- タグライブラリ, 197
- チャンネル, 189

委任管理者, 189

インポート

- 検索サーバーデータベース, 138
- ポータルデータ, 38-39

え

エクスポート, ポータルデータ, 37-38

エンドユーザー動作の追跡, 99-101

か

カテゴリ, 検索サーバー, 136-137

監視

- 概要, 103
- 設定, 104-105
- チャンネル統計, 105
- デスクトップ統計, 104, 105-106
- 無効化, 104-105

管理

- 検索サーバー, 135
- 検索サーバーデータベース, 138
- 検索サーバーロボット, 147-186
- デスクトップ, 53-73
- ポータル, 33-41
- ポータルインスタンス, 33-41
- 管理機能, 分散, 189-196
- 管理コンソール
- 概要, 28-30

管理コンソール (続き)

- テクノロジー, 28-30
- ユーザーインタフェース, 29
- ログイン, 30

け

検索サーバー

- インポートエージェントの作成, 138
 - 概要, 135-137
 - カテゴリ, 136-137
 - カテゴリの分類, 145
 - 管理, 135
 - 自動分類の属性の編集, 145-146
 - データベース, 138
 - データベースのインポート, 138
 - データベースの管理, 138
 - 分類, 136-137
 - ロボット、管理, 147-186
- 検索サーバーデータベースのエージェントのインポート, 138

こ

- コマンド行インタフェース, 31
- コンテナプロパティ, 54

さ

- サイトプローブ, 151
- 削除, 検索サーバー, 138
- 作成
 - DiscussionProvider チャンネル, 123-125
 - WSRP コンシューマ登録, 95-96
 - WSRP プロデューサ, 93-94
 - アダプタ, 130-131
 - インポートエージェント, 142-143
 - 検索サーバー, 137
 - 検索のカテゴリ, 144-145
 - 検索用のデータベース, 142
 - フィルタ、ロボット, 159-160
 - 分類ルール、ロボット, 161

作成 (続き)

- メタアダプタ, 129
- サポート関数のフィルタリング、ロボットアプリケーション関数, 169-174

し

- シミュレータ, 152
- シャットダウン関数、ロボットアプリケーション関数, 178-179
- シングルサインオンアダプタ
 - 概要, 127-128
 - メタアダプタ, 128-130

す

- スキーマ
 - データベースエイリアスの定義, 140
 - データベースの編集, 139-140

せ

- 生成関数、ロボットアプリケーション関数, 176-178

そ

- 組織
 - LDAP ディレクトリノード, 49-51
 - Portal サービスの追加, 46
 - 概要, 45
 - 新規作成, 45
 - 新規へのアクセス, 45-46
 - 必要な Portal サービスの指定, 47-48

た

- タグライブラリ、委任管理のリファレンス, 30
- タグライブラリリファレンス、委任管理, 197

ち

チャンネルおよびコンテナ

- 概要, 58
- 削除, 65-67
- 作成, 65-67
- プロパティの変更, 61-64

つ

- 追加, 設定済み WSRP プロデューサ, 97
- ツール, ロボット, 151-152

て

定義, データベーススキーマのエイリアス, 140

ディスカッション

- DiscussionLite チャンネル, 125
- DiscussionProvider, 122-123
- 概要, 121-125

ディスプレイプロファイル

- アップロード, 72-73
- 概要, 53-55
- 管理, 71-73
- グローバル, 53
- コンテナプロパティ, 54
- 削除, 73
- ダウンロード, 72
- デスクトップ属性, 55
- デフォルトインストール, 53
- 列レイアウト, 54
- ロード, 53

ディレクトリノード、LDAP

- 情報の表示, 51
- 設定方法, 49-50
- ロケーションバー, 49-51
- ロケーションバーからの削除, 51
- ロケーションバーへの追加, 50-51

データベース

- インデックスの再作成, 140
- 期限切れ, 140-141
- 検索サーバーのインポート, 138
- 検索サーバーの管理, 138
- スキーマのエイリアス, 140

データベース (続き)

- スキーマの編集, 139-140
- パーティション分割, 141
- 分析の表示, 140
- 分類, 136-137
- デスクトップ
 - 概要, 53-55
 - コンテナおよびチャンネルの管理, 58
 - コンテンツの管理, 56-68
 - 属性, 68-71
 - ディスプレイプロファイル, 53-55
 - ポートレットの管理, 56-58
- デバッグ, ロボットツール, 151-152

と

登録

- 概要, 115-116
- セットアップ, 116-121
- 匿名ユーザー, 131-132

ひ

表示

- Portal Server インスタンスの一覧, 39
- アダプタ, 129-130
- 検索レポート, 144
- データベース分析, 140
- ポータルの一覧, 35
- メタアダプタ, 129

ふ

フィルタ

- 有効化, 161
- ロボットデータの定義, 151
- フィルタリング関数, ロボットアプリケーション関数, 166-169
- 複数のポータル, 33
- ブラウザインタフェース, 29
- 分類ルール, ロボット
 - 作成, 161

分類ルール、ロボット (続き)

編集, 162

へ

編集

WSRP コンシューマ登録, 96

WSRP プロデューサ, 94-95

アダプタの設定プロパティ, 131

検索のカテゴリ, 145

データベーススキーマ, 139-140

フィルタ、ロボット, 160

リソース記述, 143-144

ほ

ポータル

一覧の表示, 35

インスタンス, 39

削除, 36

作成, 35

追加, 35

複数, 33

ポータル管理者, 知識, 19

ポートレット

概要, 56-58

現在の場所から削除, 57-58

現在の場所から配備, 56-57

設定ウィザード, 58

設定の変更, 58

ゆ

ユーザー動作の追跡

概要, 99-101

有効化, 101-102

レポートの生成, 102

ユーティリティー、ロボット, 151-152

り

リソース記述

期限切れ, 140-141

削除, 140-141

データインデックスの再作成, 140

データベースの破棄, 141

データベース分析の表示, 140

れ

列挙関数, ロボットアプリケーション関数, 175

ろ

ロギング

概要, 107

共通ロガーの設定, 109-111

結果のカスタマイズ, 109

個別ロガーの設定, 111-113

ログビューア, 108-109

ロケーションバー, 機能, 49-51

ロボット

概要, 147-152

管理, 148

起動, 152-153

サイトの定義, 153

サイトプロンプト, 155

サイトを編集, 153-154

シミュレーション, 152

巡回の制御, 154

スケジュール設定, 152

データフィルタの定義, 151

データベースの更新, 153

フィルタシミュレーション, 154-155

分類ルール, 161-162

ユーティリティー, 151-152

リソースフィルタ, 155-159

ロボットアプリケーション関数

サポート関数のフィルタリング, 169-174

シャットダウン関数, 178-179

生成関数, 176-178

セットアップ関数, 165-166

ソースとデスティネーション, 162-165

ロボットアプリケーション関数 (続き)
 フィルタリング関数, 166-169
 列挙関数, 175