



SERENA® CHANGEMAN® VERSION MANAGER™
管理者ガイド

Copyright © 1985-2006 Serena Software, Inc. All rights reserved.

本書および本書で説明しているソフトウェアは、ライセンス契約に基づいて提供されており、契約条項に従ってのみ使用および複製することができます。ライセンス契約によって許可されている場合を除き、弊社の文書による同意なく本書のいかなる部分も複製、複写、および検索システムへ保存することを禁止します。また電子、機械、記録、あるいはその他のいかなる方法によっても転送することを禁止します。複製されたドキュメントには、全部または一部であるかにかかわらず、この著作権の記述すべてが、変更されることなく含まれていなければなりません。

本書の内容は、情報の提供のみを目的としており、事前の通告なく変更されることがあり、弊社が確約するものではありません。弊社は、本書の誤記や不正確な記述について一切その責任あるいは負担を負うことはありません。

商標

Serena、TeamTrack、StarTool、PVCS、Collage、および ChangeMan は、Serena Software, Inc. の登録商標です。Serena ロゴ、Dimensions、Professional、Tracker、Version Manager、Builder、Meritage、および Mover は、Serena Software, Inc. の商標です。

その他の製品または会社名は、それらを識別する目的にのみ使用しており、それぞれの商標である可能性があります。

米国政府権利

契約者が、本契約に基づいて米国政府およびその機関のため、あるいはそれらに代わって購入したソフトウェアは、すべて FAR の定義に基づく「商用ソフトウェア」です。米国政府による使用、複製、および開示にはソフトウェア購入時に同意したライセンス契約の制約が適用されます。製造元は、Serena Software, Inc., 2755 Campus Drive, San Mateo, CA 94403 です。

資料番号：MA-VMADMIN-005

発行日：2006年8月

目次

Version Manager へようこそ	11
表記上の規則	11
マニュアルの使用法	12
技術サポート へのお問い合わせ	13

第 1 章

デスクトップクライアントを使用した Version Manager の設定	15
はじめに	15
Version Manager プロジェクトについて	15
Version Manager プロジェクト データベースについて	17
プロジェクト データベースの計画	19
作業ファイル	19
アーカイブの場所	20
作業ファイルの場所	20
共有アーカイブ	21
バージョン 5.3/6.0 プロジェクトのアップグレード	22
SourceSafe データベースの変換	23
クロスプラットフォーム環境	23
Serena Builder	24
プロジェクト データベースのセットアップ	24
プロジェクト データベースの作成	24
プロジェクトの作成	26
作業ファイルの追加	28
アーカイブの共有	32
5.3/6.0 プロジェクトのプロジェクト データベースへのコピー	33
クロスプラットフォーム環境での Version Manager のセットアップ	41
setuid の無効化	43
nfsmap ファイルの編集	43
権限の割り当て	44
Version Manager の設定	45
ファイル名およびディレクトリ名の大文字と小文字の使用	51
次のステップ	51

第 2 章

コマンドラインインターフェイスを使用した Version Manager のセットアップ	53
はじめに	53
Version Manager のコマンドラインインターフェイスについて	53
デフォルト のアーカイブ 拡張子 テンプレート	54
使用時のガイドライン	54
制限事項	54
プロジェクト 組織のシナリオ	55
小規模な単一ユーザのプロジェクト	55

大規模な単一ユーザのプロジェクト	55
ネットワークを使用した複数ユーザのプロジェクト	55
Version Manager のセットアップ計画	56
作業ファイル	56
アーカイブの場所	56
作業ファイルの場所	56
クロスプラットフォーム環境	56
ネットワークを使用した複数ユーザのプロジェクトのセットアップ	57
アーカイブディレクトリの作成	57
ローカルワークステーションでの作業ファイルディレクトリの作成	58
ローカルコンフィグレーションファイルの作成	58
作業ファイルのチェックイン	58
マスターコンフィグレーションファイルの作成	59
マスターコンフィグレーションファイル名の埋め込み	60
クロスプラットフォーム環境での Version Manager のセットアップ	60
setuid の無効化	63
nfsmap ファイルの編集	63
PVCS_BINDIR の設定	64
権限の割り当て	64
Version Manager の設定	65
ファイル名およびディレクトリ名の大文字と小文字の使用	66
次のステップ	67

第 3 章

Version Manager の設定	69
はじめに	69
コンフィグレーションオプションについて	70
デスクトップクライアントでのデフォルト設定	72
コンフィグレーションファイルが使用されない場合のデフォルト設定	74
デスクトップクライアントのコンフィグレーションファイル	75
マスターコンフィグレーションファイル	75
デスクトップクライアントでのコンフィグレーションオプション	
設定のガイドライン	76
デスクトップクライアントによるコンフィグレーションファイルの読み込み	77
コンフィグレーションファイルのプロジェクトデータベースおよび	
プロジェクトへの関連付け	77
コマンドラインインターフェイスでのコンフィグレーションファイル	79
マスターコンフィグレーションファイル	79
コマンドラインインターフェイスでのコンフィグレーション	
オプション設定のガイドライン	80
コマンドラインインターフェイスによるコンフィグレーション	
ファイルの読み込み	81
デスクトップクライアントで作成されたコンフィグレーション	
ファイルのコマンドラインインターフェイスでの使用	81
コンフィグレーションオプションの設定	81
デスクトップクライアントを使用する場合	82
コマンドラインインターフェイスを使用する場合	83
アーカイブ作成時のオプション	84
ブランチオプション	86

ロックオプション	86
プロモーションオプション	88
アクセスコントロールデータベースのオプション	88
ジャーナルファイルのオプション	89
ログインソース	90
LDAP コンフィグレーションオプション	95
作業ファイル属性のオプション	98
キーワード 展開オプション	99
参照用ディレクトリのオプション	102
アーカイブ検索パスのオプション	104
コマンドラインオプション	106
セマフォオプション	108
テンポラリファイルのオプション	110
ファイルタイプに基づいて設定されるオプション	112
イベントトリガ	116
オプションの変更許可と変更禁止	116
Version Manager へのマスターコンフィグレーションファイルの埋め込み	118
デスクトップクライアントを使用する場合	118
コマンドラインインターフェイスを使用する場合	119
TrackerLink と SourceBridge のセットアップ	120
Source Bridge ユーザオプションの設定	121
TeamTrack と TrackerLink の併用	124
Serena ChangeMan Mover の設定	125
Mover サーバへの接続の設定	125
Version Manager と Mover の併用	126
トラブルシューティング	127
コンフィグレーションファイルがほかのユーザによってロックされている	127

第 4 章

Version Manager File Server の設定と使用 129

Version Manager File Server について	130
Version Manager File Server を使用する理由	130
Version Manager File Server の互換性	131
Version Manager File Server の仕組み	132
Version Manager File Server のインストール	134
Version Manager File Server の管理について	134
File Server の起動と停止	134
Windows における File Server の起動または停止	135
UNIX における File Server の起動または停止	135
Version Manager File Server の管理ユーティリティの起動	136
管理者ユーザの管理	136
プロジェクトデータベースとリビジョンライブラリのパスの管理	137
パスマップの追加または編集	139
パスマップセキュリティオプションの設定	141
パスマップの削除	142
複数の File Server への作業の割り当て	142
File Server の設定	142
WorkDir ディレクティブと ArchiveWork ディレクティブの設定	146
サーバステータスの表示	146

Version Manager File Server の管理ユーティリティを使用した サーバステータスの表示.....	147
デスクトップクライアントを使用したサーバステータスの表示.....	148
サーバログの表示.....	148
Version Manager File Server へのプロジェクト データベースの追加.....	149
既存のプロジェクト データベースを移動せずに File Server へ 追加する方法.....	149
デスクトップクライアントを使用してプロジェクト データベースを File Server にコピーする方法.....	153
PCLI を使用してプロジェクト データベースを File Server に コピーする方法.....	155
File Server 上で新しいプロジェクト データベースを作成する 方法.....	158
リビジョンライブラリの作成.....	158
前提条件.....	159
RFSSplitOnCreate ディレクティブの有効化.....	159
既存のアーカイブの分割.....	160
アーカイブのエクスポート、インポート、移動、名前変更、および修正.....	163
構文.....	163
オプション.....	163
例.....	164
クロスプラットフォーム環境における File Server の使用.....	165
セキュリティに関する特記事項.....	166
パスマップのデフォルト アクセスコントロールデータベースの作成.....	167
File Server における Secure Socket Layer の有効化.....	167
File Server とともに使用するためのクライアントの設定.....	168
デスクトップクライアントのインストール時に File Server へのアクセスを 設定する方法.....	169
IDE クライアントのみのインストール時に File Server へのアクセスを 設定する方法.....	170
デスクトップクライアントがインストールされていない場合に Servers.ini ファイルの場所を指定する方法.....	171
Version Manager クライアントから File Server にアクセスする方法.....	172

第 5 章

環境の管理.....	173
はじめに.....	174
ワークスペースの使用.....	174
ワークスペースのタイプ.....	174
ワークスペースの階層.....	175
ワークスペースの設定の表示.....	176
パブリックワークスペースの使用.....	177
パブリックワークスペースの作成.....	179
ワークスペースの設定.....	181
カスタムツールの追加.....	182
カスタムツールの追加.....	183
ツール設定ファイルの移動.....	185
既存アーカイブの属性の変更.....	185
属性の変更が必要な場合.....	186
アーカイブ属性.....	186

ファイルタイプによって設定される属性.....	186
既存の属性の判断.....	187
デスクトップクライアントを使用する場合.....	187
コマンドラインインターフェイスを使用する場合.....	191
アーカイブの移動.....	192
アーカイブの場所の変更.....	192
アーカイブのインポート.....	193
プロジェクトとプロジェクト データベースのコピー.....	195
プロジェクトのコピー.....	195
プロジェクト データベースのコピー.....	198

第 6 章

セキュリティの使用.....	201
はじめに.....	202
アクセスコントロールデータベースについて.....	202
ユーザ.....	202
デスクトップクライアントの場合.....	203
コマンドラインインターフェイスの場合.....	204
アクセスリストについて.....	205
アクセスリスト グループ.....	205
権限について.....	206
ベース権限.....	206
コンポジット 権限.....	206
権限セット.....	207
権限のルール.....	207
セキュリティの計画.....	208
例.....	208
セキュリティの設定.....	209
デスクトップクライアントを使用する場合.....	209
コマンドラインインターフェイスを使用する場合.....	222
Version Manager へのアクセスコントロールデータベースの埋め込み.....	227
デスクトップクライアントを使用する場合.....	227
コマンドラインインターフェイスを使用する場合.....	228
プロジェクト データベース作成の制限.....	228
セキュリティのメンテナンス.....	229
アクセスコントロールデータベースからのユーザの削除.....	230
ユーザの ID の変更.....	230
ユーザ権限の変更.....	231
アクセスリスト グループ 権限の変更.....	232
アクセスリスト グループのメンバーの変更.....	233
アクセスコントロールデータベースからのアクセスリスト グループの削除.....	234
アクセスリスト の変更.....	235
セキュリティの無効化.....	235
プロジェクト データベースに関連付けられたアクセスコントロール データベースの変更.....	236
アクセスコントロールデータベースとプロジェクト データベース との関連付けの削除.....	238
権限の定義.....	238
コンポジット 権限.....	242

デフォルトの権限セット	243
-------------	-----

第 7 章	プロモーションモデルの使用	245
	はじめに	246
	プロモーションモデルのルール	246
	プロモーションモデルの定義	248
	デスクトップクライアントを使用する場合	249
	コマンドラインインターフェイスを使用する場合	251
	プロモーションとライフサイクルの管理	251
	シナリオ	251
	プロモーションと並行開発	252
	並行開発をコントロールするためのプロモーションモデルの使用	252
	シナリオ 1: 緊急のバグ修正のためのプロモーショングループの定義	253
	シナリオ 2: マルチプラットフォーム開発のためのプロモーショングループ定義	254
	ユーザによるプロモートの制限	256
第 8 章	ファイルのブランチとマージ	259
	ブランチ	260
	ブランチが作成される場合	260
	自動ブランチ	261
	自動ブランチの設定	261
	ブランチのための多重ロックの使用	266
	多重ロックの設定	266
	マージ	268
	自動マージ	269
第 9 章	イベントトリガの使用	271
	はじめに	272
	Version Manager の処理イベント	272
	カスタムツールの実行	274
	イベントトリガのセットアップ	274
	デスクトップクライアントを使用する場合	274
	コマンドラインインターフェイスを使用する場合	275
	イベントトリガへのイベント情報の転送	276
	カスタムツールのイベント情報	280
	イベント情報を渡す方法と処理の中断	281
	イベントトリガの例	286
第 10 章	レポートの使用	289
	はじめに	290
	レポートオプションの設定	290
	HTML レポートのフォーマットのカスタマイズ	292
	ジャーナルレポートの生成	293
	ジャーナルレポートの読み方	293
	デスクトップクライアントを使用する場合	294
	コマンドラインインターフェイスを使用する場合	296

履歴レポートの生成	297
履歴レポートの読み方	298
デスクトップクライアントを使用する場合	299
コマンドラインインターフェイスを使用する場合	301
セキュリティレポートの生成	302
セキュリティレポートの読み方	302
デスクトップクライアントを使用する場合	304
コマンドラインインターフェイスを使用する場合	304
プロジェクトに対する変更の表示 (change.log)	304

第 11 章

Serena Collage との統合 307

Version Manager の Serena Collage との統合について	308
統合機能の用途	308
統合機能の仕組み	308
要件	308
統合機能の設定の概要	309
Version Manager と Collage の設定	309
セットアップファイルの作成	310
ログイン情報の設定	312
Collage へのファイルの転送	313
手動によるファイルの転送	313
ファイルの自動転送	313
Version Manager デスクトップクライアントを使用したファイルの転送	314
転送したファイルの追跡	315

第 12 章

Version Manager の SourceSafe 用変換ユーティリティの使用 317

はじめに	318
ユーザ定義ラベル	318
制約	319
変換後のアーカイブの場所	319
変換ユーティリティの使用前の準備	320
VSS2VM 変換作業にかかる労力の評価	321
システムのセットアップ値と設定の検証	325
IDE プロジェクトの変換	327
Visual Studio .NET プロジェクトの変換	327
他の SCC ベース IDE プロジェクトの変換	329
変換ユーティリティの実行	330
オプション	331
国際化文字列	333
ログファイル	334
ss2pvcs.log	334
convERR.log	335
Shared.log	335
dgb_path.log	335

索引 337

Version Manager へようこそ


Serena ChangeMan Version Manager をお買い上げいただきありがとうございます。このソフトウェアは、ソフトウェアの開発手法に革命的な変化をもたらす、多機能で強力なバージョン管理システムです。Version Manager の使用により、個々のファイルへの変更の保存や追跡から開発サイクル全体の管理や監視に至るまで、ソフトウェア開発プロジェクトの各段階の構成管理および保守管理を行うことができます。

このマニュアルの内容	このマニュアルでは、デスクトップクライアントまたはコマンドラインインターフェイスを使用して、Version Manager のプロジェクトやアーカイブを設定、管理、および維持する方法について説明します。
詳細について	Version Manager のマニュアルセットについて、Version Manager の基本的な操作方法、およびオンラインヘルプの使用方法については、『Serena ChangeMan Version Manager 入門ガイド』を参照してください。
対応バージョン	このマニュアルの内容は <i>Serena ChangeMan Version Manager</i> バージョン 8.1.3（日本語版）以降を対象としています。この版によって、このマニュアルの旧版は廃版となります。

表記上の規則

オンラインマニュアルおよびオンラインヘルプでは、以下の表記規則を採用しています。ドキュメントを使用する場合は、これらの表記規則を参考にしてください。ただし、各種製品コンポーネントおよびホストオペレーティングシステムについては、それぞれの表記規則に従ってください。

表記例	意味
斜体	ドキュメント内でまだあまり使われていない新しい用語を紹介し、時として強調を示すために使われます。
太字	重要な情報やフィールド名を強調します。
大文字	使用可能なキーまたはキーの組合せを示します。例：ENTER キーを押します。
固定幅フォント	構文例、ユーザ指定が可能な値、システムから返される結果値などを表します。
固定幅斜体	ユーザが特定の値を指定する箇所を表します（例： <i>filename</i> ）。
固定幅太字	実行されたコマンドの結果を表します。
横棒（-）と縦棒（ ）	メニューとメニューに関連するコマンドを区切ります。例：[ファイル] [コピー] を選択は、[ファイル] メニューの [コピー] を選択することを意味します。 また、縦棒（ ）は、構文のコマンドラインで使用され、棒の左右の値のうちどちらかを指定しなければならないことを示しています。
角括弧 []	オプション項目を示します。オプションの項目を表します。たとえば、SELECT [DISTINCT] という文で、DISTINCT はオプションのキーワードです。

表記例	意味
...	複数の値を持つ場合があるコマンド引数を表します。
	クリックするショートカットアイコンを示します。ショートカットアイコンは余白に置かれます。

マニュアルの使用法

Serena® ChangeMan® Version Manager™ マニュアルは、Adobe PDF 形式を使用しています。PDF ファイルの表示には、Adobe® Reader® を使用してください。これは、www.adobe.com/jp から無償で入手することができます。



注 必ず、Adobe Reader のフルバージョンをダウンロードしてください。基本バージョンには検索機能がありません。

ここでは、Adobe Reader の主な機能をいくつか説明します。Adobe Reader の詳細については、Adobe Reader のオンラインヘルプを参照してください。

PDF 形式のマニュアルには次の機能があります。

- **しおり**：すべてのマニュアルには、特定の項目に移動できるしおりが用意されています。デフォルトでは、しおりは各オンラインマニュアルの左側に表示されます。
- **リンク**：マニュアル内の相互参照リンクにより、マウスをクリックするだけでマニュアル内の他のページあるいは他のマニュアルへ移動できます。リンクは青で表示されています。
- **印刷**：マニュアルを参照中に、現在のページ、ページ範囲、あるいはマニュアル全体を印刷することができます。
- **高度な検索**：Adobe Reader のバージョン 6 以降では、指定ディレクトリ内の複数の PDF ファイルから検索できる高度な検索機能が用意されています。次の手順を参照してください。

一度に複数の PDF 文書から検索するには、次の手順に従います（Adobe Reader バージョン 6 以降）。

- 1 Adobe Reader から、[編集] - [検索] を選択します（あるいは SHIFT+CTRL+F を押します）。
- 2 テキストボックス内に検索する単語または文を入力します。
- 3 〈以下の場所にあるすべての PDF 文書〉オプションを選択し、検索するフォルダを参照します。
- 4 必要に応じて、〈完全に一致する語のみ〉、〈大文字と小文字を区別〉など、その他のオプションを選択します。
- 5 【検索】をクリックします。



注 さらに強力な検索オプションを使用するには、Reader ウィンドウの右下の〈高度な検索オプションを使用〉リンクをクリックします。（リンクが〈基本的な検索オプションを使用〉になっている場合はすでに高度な検索オプションが有効になっています。）詳細については、Adobe Reader のオンラインヘルプを参照してください。

技術サポートへのお問い合わせ

登録ユーザは、<http://support.serena.com/> にログインすることができます。

第 1 章

デスクトップクライアントを使用した Version Manager の設定

はじめに	15
Version Manager プロジェクトについて	15
Version Manager プロジェクトデータベースについて	17
プロジェクトデータベースの計画	19
プロジェクトデータベースのセットアップ	24
クロスプラットフォーム環境での Version Manager のセットアップ	41
次のステップ	51

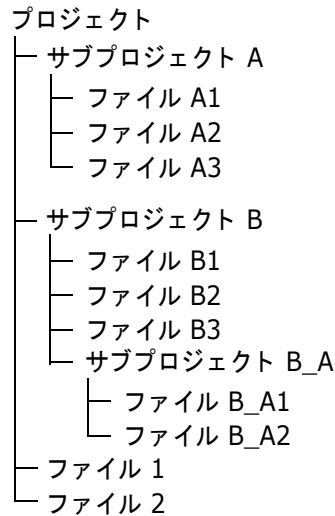
はじめに

デスクトップクライアントを使用した Serena ChangeMan Version Manager のセットアップ手順を理解するには、Version Manager のプロジェクトデータベースとプロジェクトに関する理解が必要です。この章では、デスクトップクライアントを使用して Version Manager プロジェクトデータベースの計画と定義を行うときに役立つ情報を記載しています。

コマンドラインインターフェイスを使用した Version Manager のセットアップについては、[53 ページの『コマンドラインインターフェイスを使用した Version Manager のセットアップ』](#)を参照してください。

Version Manager プロジェクトについて

プロジェクトとは、Version Manager 内に保存する互いに関連したファイルのセットで、Version Manager はこれを使用してファイルの整理を行います。プロジェクト内には、次の図に示すようにサブプロジェクトとファイルを配置できます。



Version Manager プロジェクトは、ファイルシステムのディレクトリに似ています。ディレクトリと同様、プロジェクトは作成とメンテナンスの可能なファイルのセットになっています。また、ディレクトリと同じくプロジェクトは階層構造になっており、プロジェクトやほかのサブプロジェクトの下にはサブプロジェクトが存在することがあります。ただし、プロジェクトではディレクトリと異なり、古いバージョンのファイルがアーカイブ中にデルタファイルとして保存されます。ディレクトリでは古いバージョンのファイルは保存されません。



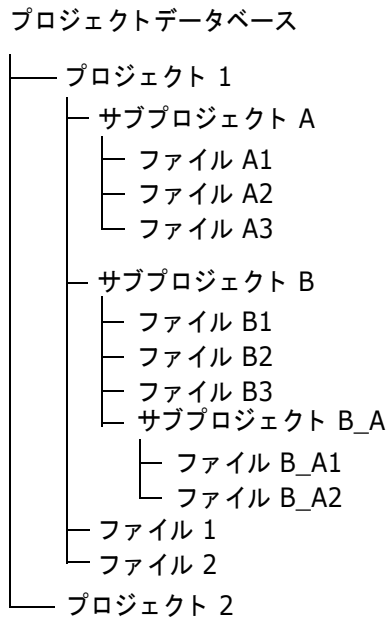
注 変更ログ

サブプロジェクトやバージョン管理ファイルの追加や削除など、プロジェクトの内容に対する変更は、プロジェクトデータベースのルートディレクトリにある `change.log` ファイルに記録されます。詳細については、[304 ページの『プロジェクトに対する変更の表示 \(change.log\)』](#)を参照してください。

Version Manager プロジェクトは、すべて Version Manager プロジェクトデータベースに格納されます。

Version Manager プロジェクトデータベースについて

プロジェクトデータベースは、プロジェクトのための格納場所となります（下図参照）。



プロジェクトデータベースには、そのプロジェクトデータベース内にあるすべてのプロジェクトに対し、Version Manager がどのように動作するかが定義されています。デフォルトでは、プロジェクトの設定はプロジェクトデータベースから引き継がれます。設定はプロジェクトごとに変更できます。

Version Manager を最初に実行するときには、空（から）のプロジェクトデータベースが開かれるようにすることができます。このプロジェクトデータベースはすでに設定されていますが、プロジェクトはまったく含まれていません。この空のプロジェクトデータベースは、Version Manager がインストールされたディレクトリの下に newdb というディレクトリに作成されます。



注 プロジェクトデータベースの作成については、システム管理者により権限が与えられていない場合があります。この権限が与えられていない場合は、最初に Version Manager を実行したときに新しい空のプロジェクトデータベースは作成されません。

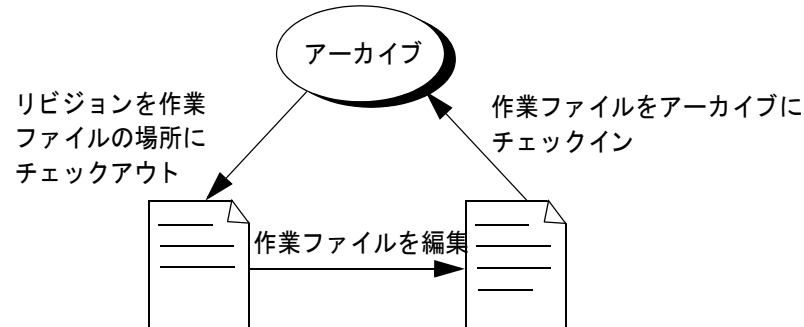
プロジェクトデータベースの作成時に作成されるデフォルトのディレクトリとファイルは次のとおりです。

- archives ディレクトリ
- コンフィグレーションファイル
- アクセスコントロールデータベース
- work ディレクトリ
- pvcsuser ディレクトリ
- ツールコンフィグレーションファイル
- lib ディレクトリ
- システム識別ファイルとメタデータファイル

archives ディレクトリ	archives ディレクトリにはまだ、アーカイブファイルは含まれていません。アーカイブファイルはプロジェクトデータベース内にプロジェクトを作成し、そのプロジェクトにファイルを追加したときに配置されます。このディレクトリの場所は、プロジェクトデータベースを作成するとき指定します。
コンフィグレーション ファイル	<p>デフォルトでは、archives ディレクトリに新しいコンフィグレーションファイルが作成されます。セキュリティ上の理由により、このコンフィグレーションファイルの名称は暗号化されて表示されません。たとえば、「c6bonpj1.cfg」のような名称が設定されます。このコンフィグレーションファイルには、プロジェクトデータベース内のすべてのプロジェクトに対するコンフィグレーション設定が記録されます。</p> <p>Version Manager はこのファイルをほかのどのコンフィグレーションファイルよりも先に読み込みますが、これにより確実に、すべてのユーザに同一のオプション設定を共有させることができます。このファイル中の設定は、必要に応じて変更できます。また、必要に応じて、プロジェクトデータベースの作成時に既存のコンフィグレーションファイルを使用したり、既存のコンフィグレーションファイルをデフォルトの場所（archives ディレクトリ）にコピーするように指定することもできます。コンフィグレーションファイルの詳細については、75 ページの『デスクトップクライアントのコンフィグレーションファイル』を参照してください。</p>
アクセスコントロール データベース	<p>デフォルトでは、archives ディレクトリに新しいアクセスコントロールデータベースが作成されます。このファイルは、コンフィグレーションファイル中では使用しないように設定されています。セキュリティ上の理由により、このアクセスコントロールデータベースの名称は暗号化されて表示されません。たとえば、「c6bonpj1.db」のような名称が設定されます。</p> <p>アクセスコントロールデータベースではプロジェクトに対して作業を行うことが承認されたユーザが定義され、またそのユーザが行える作業が定義されます。また、必要に応じて、プロジェクトデータベースの作成時に既存のアクセスコントロールデータベースを使用するか、既存のアクセスコントロールデータベースをデフォルトの場所（archives ディレクトリ）にコピーするように指定することもできます。アクセスコントロールデータベースの詳細については、202 ページの『アクセスコントロールデータベースについて』を参照してください。</p>
work ディレクトリ	work ディレクトリは空で、Version Manager アーカイブからリビジョンをチェックアウトするときのデフォルトの場所となります。このディレクトリの場所は、プロジェクトデータベースの作成時に指定します。
pvcsuser ディレク トリ	pvcsuser ディレクトリには、プライベートワークスペースの設定などのユーザ情報が含まれます。
ツールコンフィグ レーションファイル	ツールコンフィグレーションファイルには、Serena ChangeMan Dimensions がシステムにインストールされている場合、PVCS TeamLink を起動するためのカスタムツールが 1 つ含まれています。このファイルにはカスタムツールを追加できます。 182 ページの『カスタムツールの追加』 を参照してください。
lib ディレクトリ	lib ディレクトリには、プロジェクトデータベースの作成に使用した Version Manager のバージョン（リリース）に関する情報が含まれています。
メタデータファイル	ファイルなどを識別するための pvcsid.ser という名称のファイルが作成されます。このファイルの移動や削除は行わないでください。また、pvcsroot.ser および pvcsproj.ser というファイルが作成され、使用されるコンフィグレーションファイルおよび、作業ファイルやアーカイブの場所などに関するメタデータが含まれる場合があります。

プロジェクトデータベースの計画

アーカイブが含まれる Version Manager プロジェクトデータベースは、通常はネットワーク上のファイルシステムに置かれ、開発者はファイルに対する作業をローカルのマシン上で行います。この作業の流れは次のようになります。



プロジェクトデータベースを定義するには、次の情報が必要となります。

- ソース管理を行う作業ファイル（ソースファイル）をどこに置くか。19 ページの『作業ファイル』を参照してください。
- ファイルのアーカイブをどこに置くか。20 ページの『アーカイブの場所』を参照してください。
- デフォルトで、どこでユーザは作業ファイルを編集するか。作業ファイルの場所は、多くの場合にはファイルが最初に置かれていたディレクトリです。20 ページの『作業ファイルの場所』を参照してください。
- ソース管理を行うファイルのいずれかが、2 つ以上のプロジェクトまたはサブプロジェクトによって共有されているかどうか。21 ページの『共有アーカイブ』を参照してください。
- 新バージョンの Version Manager にアップグレードすべきバージョン 5.3/6.0 の Version Manager アーカイブとプロジェクトが存在するかどうか。22 ページの『バージョン 5.3/6.0 プロジェクトのアップグレード』を参照してください。
- 変換すべき SourceSafe のファイルが存在するかどうか。23 ページの『SourceSafe データベースの変換』を参照してください。
- 使用する環境がクロスプラットフォーム環境かどうか。23 ページの『クロスプラットフォーム環境』を参照してください。
- プロジェクトデータベースを Serena Builder と統合するかどうか。24 ページの『Serena Builder』を参照してください。

作業ファイル

プロジェクトデータベースをセットアップするときは、作業ファイルのディレクトリ構造をそのまま採用してください。たとえば、次のディレクトリ構造があるとする、MyApp という名称のプロジェクトデータベースと、その下に include、resource、および source の 3 つのプロジェクトを定義することをお勧めします。



デフォルトでは、Version Manager は作業ファイルの構造を反映したアーカイブの場所を自動的に作成します。これらの場所の定義は、プロジェクトデータベースの作成時に変更することができます。

アーカイブの場所

アーカイブの場所とは、作業ファイルのための新規アーカイブが置かれる場所です。通常は、この場所は、許可されたすべてのユーザがアクセスできる、ネットワーク上のファイルシステム上に指定されます。デフォルトではアーカイブの場所はプロジェクトデータベースの下のディレクトリになりますが、ほかの場所もプロジェクトデータベースの作成時に指定できます。



注 Version Manager File Server を使用している場合は、アーカイブはサーバ上にある可能性があります。129 ページの『Version Manager File Server の設定と使用』を参照してください。

UNIX ユーザのみ : Version Manager ファイルは、アーカイブにさらにセキュリティレベルを追加するため、setuid モードでインストールされます。setuid ではユーザは自分自身としてログインしますが、Version Manager は実行ファイルを所有するユーザとしてパブリックアーカイブを作成します。この目的のため、pvcs という名称のユーザを作成することをお勧めします。これがアーカイブとの読み込みと書き込みを行える唯一のユーザとなります。



注 Version Manager File Server を使用している場合、setuid を使用してアーカイブを保護する必要はありません。129 ページの『Version Manager File Server の設定と使用』を参照してください。

setuid モードで実行するときには、作業ファイル、一時使用ファイル、pvcspriv プロジェクトドメイン、および \$HOME/.islvrc を除き、すべてのファイルはユーザ pvcs のもとで作成されます。アクセスコントロールの権限は、Version Manager のアクセスコントロールデータベースによってコントロールされます (201 ページの『セキュリティの使用』を参照)。pvcs でない個々のユーザは、Version Manager のコマンドを使用しているときにのみ、ファイルまたはディレクトリの修正、追加、または削除を行うことができます。

setuid モードで実行していない場合は、アーカイブディレクトリへの許可を持っている任意のユーザが、基本的な UNIX コマンドを使用してアーカイブの移動、名称変更、破壊、および削除を行うことが可能となります。



注 クロスプラットフォーム環境内で実行している場合は、setuid を使用できる可能性があります。クロスプラットフォーム環境における setuid の設定については、『Serena ChangeMan Version Manager インストールガイド』を参照してください。setuid を無効にする方法については、43 ページの『setuid の無効化』を参照してください。

作業ファイルの場所

作業ファイルの場所とは、アーカイブからチェックアウトされたリビジョンが置かれる場所です。通常はアーカイブの場所はプロジェクトデータベースの下のディレクトリになりますが、どの場所でもプロジェクトデータベースの作成時に指定できます。リビジョンをチェックアウトすると、それが作業ファイルになります。

プロジェクトデータベース中のそれぞれのプロジェクトにも、作業ファイルの場所があります。これは、たとえば c:¥prjdb¥work¥project1、c:¥prjdb¥work¥project2、のように指定されます。

プロジェクトデータベースとそのプロジェクトの作業ファイルの場所は、ワークスペース内に保存されます。ワークスペースは、パブリックまたはプライベートのいずれかで使用されます。それぞれのプロジェクトデータベースごとに、デフォルトのパブリックワークスペース（「ルートワークスペース」という名前の）が作成されます。

管理者はほかのパブリックワークスペースを作成し、それが元の作業ファイルの場所に優先するよう設定できます。たとえば、承認を得たすべてのユーザによってアクセス可能な、ネットワーク上のファイルシステム上に作業ファイルの場所を定義するよう、パブリックワークスペースを作成できます。これにより、ユーザがデフォルトの作業ファイルの場所にアクセスできることが保証されます。

次にユーザは、ファイルに対する作業をローカルで行えるよう、作業ファイルの場所をローカルのハードドライブに変更して、プライベートワークスペースを作成できます。あるいは共通して使用されるローカルドライブを、デフォルトの作業ファイルの場所として指定できます（例：Windows では C:）。ワークスペースの作成については、179 ページの『パブリックワークスペースの作成』を参照してください。

既存のファイルに対してプロジェクトをセットアップするときは、通常はその既存のファイルがあるディレクトリを作業ファイルの場所とします。

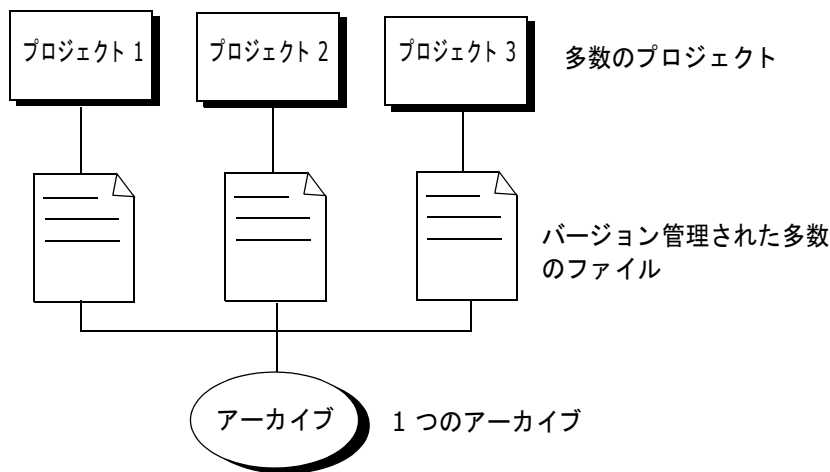
作業ファイルの場所には、そのパスのルートにある `$HOME` を含めることができます（例：UNIX 上で `$HOME/work` は `/usr/s/cherylc/work` に展開できます）。HOME 環境変数の値が置換され、作業ファイルの場所が決定されます。`$HOME` の使用により、ユーザは、自分の HOME 環境変数の値に応じて、自動的に自分用のパスを定義することができます。



注 ルートワークスペースで作業中に、プロジェクトの作業ファイルの場所を削除すると、そのプロジェクト用の作業ファイルの場所はプロジェクトデータベースディレクトリの直下に設定されます。

共有アーカイブ

Version Manager では、複数のプロジェクトがアーカイブを共有することができます。この場合は、次の図に示すように、ファイルを保存するアーカイブを 1 つ設定し、多数のプロジェクトにそのファイルを参照させます。



たとえば、ある企業が、同一のライセンス契約を使用する 3 種類のソフトウェア製品を開発、販売するとします。このライセンス契約は 3 つの製品すべてで共有することができます。この場合は、ライセンス契約のためのアーカイブを 1 つのプロジェクト内に作成し、ほかの 2 つのプロジェクトにこのアーカイブを参照させます。

アーカイブを共有すると、各製品のライセンス契約が同一であることが保証されます。その反対に、共通ファイルのコピーを複数作成して、それぞれのプロジェクトごとのアーカイブに保存すると、1つが更新されたが、ほかのものが更新されなかったというような、ファイルの不一致が発生することが防止されます。

複数のプロジェクトの間でファイルを共有するには、共有されるすべてのファイルに対してプロジェクトを1つ設定し、共有されるすべてのファイルに対するアーカイブをこのプロジェクトに含めます。次にバージョン管理ファイルをほかのプロジェクトにコピーします。ほかのプロジェクトは、コピーされたファイルのアーカイブの実体を含まず、元のプロジェクトのアーカイブを参照します。

プロジェクトを定義したら、[編集]メニューの[コピー]を選択していつでもファイルをコピーできます。バージョン管理ファイルのコピーが、そのアーカイブを参照するコピー先プロジェクトに作成されます。

バージョン 5.3/6.0 プロジェクトのアップグレード

既存の Version Manager 5.3/6.0 プロジェクトに対して、新しい Version Manager の全機能を使用するには、そのプロジェクトを新しいフォーマットの Version Manager プロジェクトデータベースにコピーする必要があります。

Version Manager Development Interface を使用して作成した 5.3/6.0 プロジェクトをアップグレードすることもできます。Version Manager Development Interface で作成された 5.3/6.0 プロジェクトをアップグレードする方法については、『Serena ChangeMan Version Manager IDE クライアント操作ガイド』を参照してください。

プロジェクトをコピーせず、Version Manager から 5.3 プロジェクトを開いた場合でも、そのプロジェクトは認識および表示されますが、Version Manager の機能のすべてを使用することはできません。ただし、次のものを含めて、大半の機能はサポートされます。

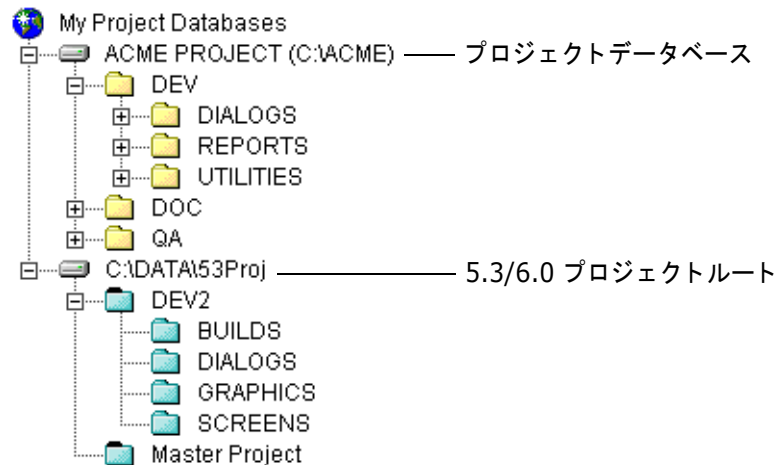
- フォルダの作成と削除
- アーカイブとのファイルのチェックインおよびチェックアウト
- 作業ファイルの追加（ただし、ディレクトリ構造は追加できない）
- ワークスペースの定義と設定
- リビジョンの削除
- プロモーションモデルとバージョンラベルの使用
- アーカイブのインポート（ただし、ディレクトリ構造は追加できない）

サポートされない機能は次のとおりです。

- 5.3/6.0 プロジェクトルート内での新しいプロジェクトの作成
- 5.3/6.0 プロジェクトルートからのプロジェクトの削除
- 5.3/6.0 プロジェクトルート内でのサブプロジェクトの作成
- 新しいフォーマットのプロジェクトデータベースから 5.3/6.0 プロジェクトルートへの、プロジェクト、サブプロジェクト、およびバージョン管理ファイルのコピー
- 5.3/6.0 プロジェクトルートの設定
- アクセスコントロールデータベース内でのセキュリティ設定の変更

新規プロジェクトデータベースでは名称とパス名が表示されますが、Version Manager で 5.3/6.0 プロジェクトルートを開いた場合は、パス名のみがプロジェクト表示部に表示されます。また、5.3/

6.0 プロジェクトルートを示すアイコンの色は青となり、新しいフォーマットのプロジェクトデータベースのアイコンは黄色いフォルダで表示されます。



SourceSafe データベースの変換

Visual SourceSafe データベースのファイルを Version Manager アーカイブに変換することができます。変換により次の動作が実行されます。

- 変換する各 SourceSafe ファイルに関する履歴とコメントを保持する。
- 変換中の SourceSafe プロジェクトに対応した Version Manager プロジェクトを作成する。
- ユーザにより定義されたラベルを保持する (318 ページの『ユーザ定義ラベル』参照)。
- ブランチを変換する。
- 変換中に見つかった共有ファイルを使用するようプロジェクトを設定する。

SourceSafe ファイルの Version Manager アーカイブへの変換の詳細については、317 ページの第 12 章『Version Manager の SourceSafe 用変換ユーティリティの使用』を参照してください。

クロスプラットフォーム環境

Version Manager デスクトップクライアントを使用することにより、Windows と UNIX の間でコンフィグレーションファイル、アクセスコントロールデータベース、プロジェクトデータベース、およびアーカイブを共有できます。プロジェクトデータベースを経由して UNIX に保存されたアーカイブに Windows からアクセスでき、またその逆も可能です。

File Server 対応のクロスプラットフォーム環境

Version Manager File Server は、OS にかかわらず、Version Manager のアーカイブやクライアントと連動します。クロスプラットフォーム環境を有効にする際、特別な手順は一切不要です。

Version Manager File Server の詳細については、129 ページの『Version Manager File Server の設定と使用』を参照してください。

ネットワーク対応のクロスプラットフォーム環境

ネットワーク対応のクロスプラットフォーム環境でプロジェクトデータベースにアクセスするには、Windows 環境の Version Manager デスクトップクライアントからそのプロジェクトデータベース

を作成する必要があります。Windows で作成されたプロジェクトデータベースには（それが UNIX 環境にあるか、Windows 環境にあるかにかかわらず）Windows と UNIX の両方からアクセスできますが、UNIX で作成されたプロジェクトデータベースには UNIX 環境の Version Manager デスクトップクライアントからのみアクセスできます。



注 クロスプラットフォーム環境内で実行している場合は、setuid を使用できる可能性があります。クロスプラットフォーム環境における setuid の設定については、『Serena ChangeMan Version Manager インストールガイド』を参照してください。

Serena Builder

Version Manager データベースで Serena Builder を使用するには、プロジェクトデータベースへのパスにドル記号 (\$) が含まれないことを確認する必要があります。これは Builder との統合の場合にのみ必要です。

Serena Builder の詳細については、『Serena ChangeMan Builder for Professional ユーザガイド』を参照してください。

プロジェクトデータベースのセットアップ

次に、Version Manager プロジェクトデータベースをセットアップするための基本的手順の概要を示します。詳細な手順については、以降のセクションで説明します。

- 1 プロジェクトデータベースを作成します。これはそのプロジェクトの存在するアーカイブを定義します。プログラムを最初に実行するときに、自動的に空のプロジェクトデータベースが作成されるように Version Manager を設定できます。新しくプロジェクトデータベースを作成するかわりに、この空のプロジェクトデータベースを使用することもできます。
- 2 必要に応じて、データベース内にプロジェクトを作成します。これはプロジェクト中のファイルに対する作業ファイルの場所を定義します。

既存のファイルに対してプロジェクトをセットアップするときは、プロジェクトを作成する必要はありません。単にディレクトリをプロジェクトデータベースに追加すれば、自動的にプロジェクトが作成されます。個別のファイルをプロジェクトに追加する場合および、ファイルを現在の作業ファイル構造以外の方法で整理する場合にのみ、プロジェクトを作成してください。
- 3 作業ファイルをプロジェクトデータベースまたはプロジェクトに追加します。
- 4 必要に応じて、ファイルを、アーカイブを共有するそれぞれのプロジェクトに追加します。
- 5 必要に応じて、5.3/6.0 プロジェクトを新しいフォーマットのプロジェクトにアップグレードします。

プロジェクトデータベースの作成

プログラムを最初に実行するときに、自動的に空のプロジェクトデータベースが作成されるようにすることができます。作業を開始するには、プロジェクトデータベースを新しく作成することも、あるいはこのプロジェクトデータベースを使用することも可能です。

プロジェクトデータベースを作成するには

- 1 [管理] - [プロジェクトデータベースの作成] を選択します。[プロジェクトデータベースの作成] ダイアログボックスが表示されます。

- 2 <名前> フィールドにプロジェクトデータベースの名前を入力します。

名前には、最初と最後にタブおよびスペースを使用できないこと以外に、使用できる文字の制限はありません。

- 3 <場所> フィールドで、プロジェクトデータベースの場所を指定します。

この場所は、このプロジェクトデータベースにアクセスするすべてのユーザからアクセス可能な場所であることが必要です。プロジェクトデータベースは、ほかのプロジェクトデータベースの下に作成することはできません。したがって、ドライブのルートレベルにはプロジェクトデータベースを作成しないことをお勧めします。



注

- プロジェクトデータベースは既存の 5.3/6.0 プロジェクトルート以外の場所に作成してください。
- Serena Builder と統合する場合、パスにドル記号 (\$) を入れることはできません。24 ページの『Serena Builder』を参照してください。

- 4 アーカイブを、デフォルト以外の場所に置く場合は、<アーカイブの場所> フィールドに場所を指定します。

デフォルトの場所は、プロジェクトデータベースの場所の下にある archive というディレクトリです。これは、アーカイブにアクセスする全ユーザがアクセス可能な場所である必要があります。

- 5 <作業ファイルの場所> フィールドに作業ファイルの場所を指定します。このフィールドは必ず指定しなければなりません。

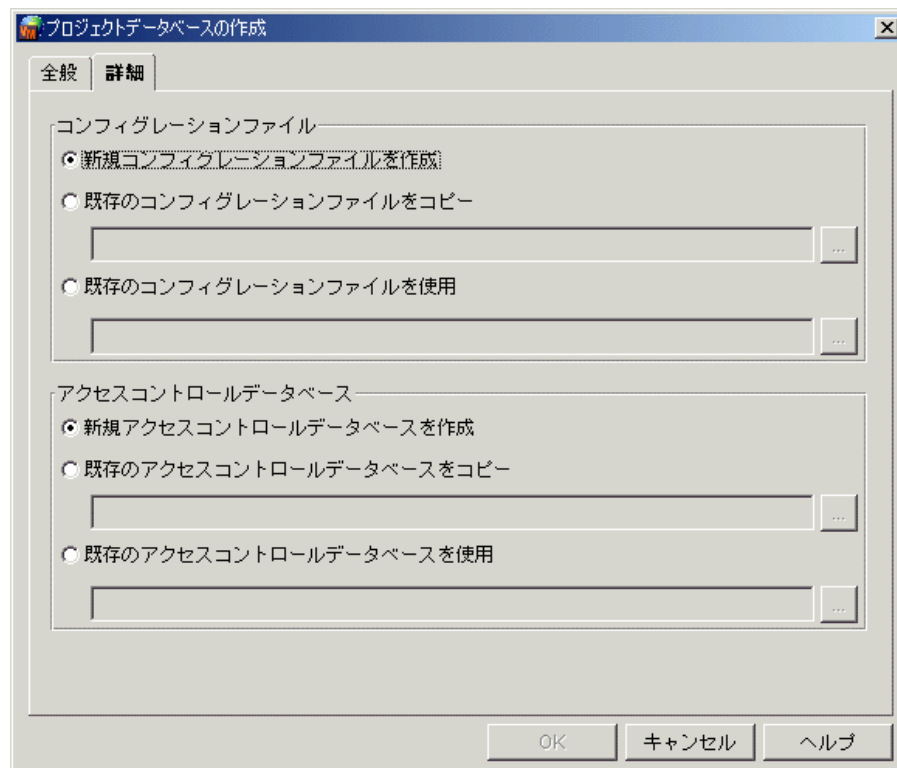
UNIX では、作業ファイルの場所にパスのルートとして \$HOME を含めることができます（たとえば、\$HOME/work は /usr/cherylc/work に展開できます）。Version Manager は、作業ファイルの場所を判断するときに、HOME 環境変数の値を置き換えます。\$HOME を使用することにより、パスは自動的にユーザの HOME 変数の値に応じて個別に設定されます。ユーザの HOME 変数の値が定義されていない場合は、作業ファイルの場所を判断するときにスペースで置き換えられます。



注 ほとんどのユーザにアクセス可能な、ネットワーク上のドライブを指定することを推奨します。ユーザは必要に応じて、作業ファイルの場所を変更するためのプライベートなワークスペースをローカルドライブに作成することができます。

- 6 必要に応じて、[詳細] タブをクリックし、このプロジェクトデータベースに関連するコンフィグレーションファイルとアクセスコントロールデータベースを指定します。

指定しない場合は、Version Manager により、プロジェクトデータベースに対する新規コンフィグレーションファイルおよびアクセスコントロールデータベースが作成されます。



- 7 【OK】 をクリックします。

プロジェクトデータベースの場所に、いくつかのディレクトリとファイルが作成されます。作成されるディレクトリとファイルについては、[17 ページの『Version Manager プロジェクトデータベースについて』](#)を参照してください。

プロジェクトの作成

既存のファイルに対してプロジェクトをセットアップするには、プロジェクトを作成する必要はありません。その代わりに [ファイル] から [作業ファイルの追加] を選択してディレクトリとファイル

をプロジェクトデータベースに追加すれば、プロジェクトが自動的に作成されます。この場合、プロジェクトはディレクトリと同じ名称で作成され、そのプロジェクトにディレクトリ内のファイルが追加されます。

ディレクトリにサブディレクトリがある場合は、必要に応じて Version Manager にサブプロジェクトを自動的に作成させ、そのサブプロジェクトに該当するファイルを含めることができます。

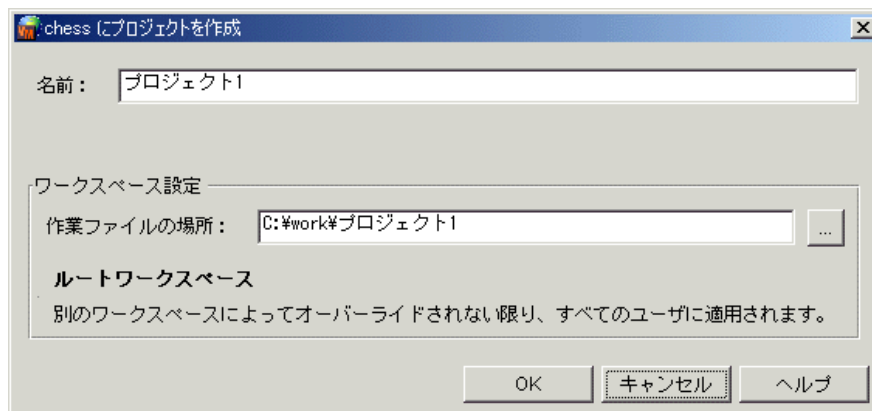
バージョン管理ファイルを現在の作業ファイル構造以外の方法で整理する場合、または個別のファイルをプロジェクトに追加する場合は、プロジェクトを作成することもできます。

プロジェクトの作成時には、以下のディレクトリが作成されます。

- プロジェクトデータベースのアーカイブディレクトリの下にプロジェクトのアーカイブディレクトリ（例：¥myapps¥ProjectDB1¥archives¥ProjectA）
- 指定した場所にプロジェクトの作業ディレクトリ。デフォルトでは、プロジェクトデータベース作業ディレクトリの下です。（例：¥myapps¥ProjectDB1¥work¥ProjectA）
- プロジェクトデータベースの場所の下にプロジェクトのメタデータディレクトリ。このディレクトリ名は、セキュリティ上の理由により暗号化されて表示されます（例：p1otxdt.prj）。プロジェクトに作業ファイルを追加すると、ここに pvcsproj.ser というファイルが 1 つ作成されます。このファイルには、サブプロジェクト、バージョン管理ファイル、コンフィグレーションファイル、アーカイブと作業ファイルの場所などのプロジェクトメタデータが記録されます。

プロジェクトを作成するには

- 1 プロジェクトを作成するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [ファイル] - [プロジェクトの作成] を選択します。[プロジェクトの作成] ダイアログボックスが表示されます。



- 3 〈名前〉フィールドで、このプロジェクトを識別できる名称を指定します。

プロジェクトデータベース中の同一レベルでは、それぞれのプロジェクトに異なる名称が必要です。この名称の最初と最後には、タブまたはスペースは使用できません。名称にはアスタリスク (*)、コロン (:), 縦棒 (|), スラッシュ (/), 円記号 (¥), 疑問符 (?), および山かっこ (< >) 以外のすべての文字を使用できます。ピリオド 2 つ (..) または 1 つ、または @ のみの名称を付けることはできません。

- 4 作業ファイルをデフォルト設定以外の場所に置く場合は、〈作業ファイルの場所〉フィールドで場所を指定します。

デフォルトの場所は、この新しいプロジェクトが属するプロジェクトまたはプロジェクトデータベースの、作業ファイルの場所の下にあるサブディレクトリです。たとえば、¥myapps¥ProjectDB1 にあるプロジェクトデータベースの下に ProjectA という名称のプロ

ジェクトを作成する場合は、この新しいプロジェクトのデフォルトの作業ファイルの場所は `¥myapps¥ProjectDB1¥work¥ProjectA` になります。

UNIX では、作業ファイルの場所にはパスのルートとして `$HOME` を含めることができます（たとえば、`$HOME/work/projecta` は `/usr/cherylc/work/projecta` に展開できます）。Version Manager は、作業ファイルの場所を判断するときに、HOME 環境変数の値を置き換えます。

`$HOME` を使用することにより、パスは自動的にユーザの HOME 変数の値に応じて個別に設定されます。ユーザの HOME 変数の値が定義されていない場合は、作業ファイルの場所を判断するときにスペースで置き換えられます。



注 ほとんどのユーザがアクセスできるネットワーク上のドライブを指定してください。ユーザは、作業ファイルの場所を変更するためのプライベートなワークスペースをローカルドライブに作成することができます。

作業ファイルの追加

プロジェクトデータベースまたはプロジェクトには、ディレクトリツリー、単独のディレクトリ、または個別のファイルを追加できます。ディレクトリを追加するときには、次の作業が自動的に行われます。

- ディレクトリと同じ名称のプロジェクトを作成し、ディレクトリ中のファイルをプロジェクトに追加する。
- ディレクトリ中の各ファイルについてアーカイブを作成する。
- 新しいアーカイブを参照する、バージョン管理ファイルを作成する。
- サブディレクトリが追加されたときにサブプロジェクトを作成し、そのサブディレクトリ中のファイルをサブプロジェクトに追加する。

デフォルトでは、作業ファイルがチェックインされたときに、「最初のリビジョン」というコメントが使用されます。このコメントは、[作業ファイルの追加] ダイアログボックスからは変更できません。

Issue の関連付け

Tracker TrackerLink または TeamTrack SourceBridge をインストールしている場合は、[作業ファイルの追加] ダイアログボックスから【Issue の関連付け】を使用して、TrackerLink または SourceBridge から Issue（不具合、変更要求、タスクなど）を作業ファイルに関連付けることができます。チェックイン時に Issue との関連付けを義務付けるために TrackerLink または SourceBridge を自動的に起動するよう Version Manager を設定することができます。[120 ページの『TrackerLink と SourceBridge のセットアップ』](#)を参照してください。

初期デフォルト値

プロジェクトに作業ファイルまたはディレクトリを追加するときには、次のデフォルト設定が使用されます。

- 選択されたディレクトリからプロジェクトの作業ファイルの場所へ、作業ファイルをコピーする。
- サブディレクトリのあるディレクトリを追加する場合は、ファイルをサブディレクトリに追加する。
- プロジェクト中に同じ名称のバージョン管理ファイルが存在する場合は、作業ファイルを追加せず、バージョン管理ファイルがすでに存在するという警告を表示する。
- 作業ファイルの読み取り専用のコピーを保持する。つまり、ファイルの追加後に作業ファイルを削除しないか、またはリビジョンをロックされたままにします。
- バージョンラベルを割り当てないか、またはブランチを作成しない。

- 作業ファイルをアーカイブにチェックインする。このデフォルト設定は、SuperUser の権限を持っている場合にのみ変更できます。持っていない場合は、デフォルト設定以外では使用できません。



注 5.3/6.0 プロジェクトに作業ファイルを追加している場合は、〈アーカイブの場所〉リストボックスから、作業ファイルのアーカイブを保管するアーカイブディレクトリを選択する必要があります。

作業ファイルを追加するには

- 1 作業ファイルを追加するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [ファイル] - [作業ファイルの追加] を選択します。[作業ファイルの追加] ダイアログボックスが表示されます。

- 3 [全般] タブで以下の操作を行います。
 - a 〈追加される作業ファイルの場所〉フィールドに追加する作業ファイルのある場所を入力するか、[...] をクリックしてその場所を選択します。



注 * または *.* 以外のフィルタを持つパス（たとえば c:\files*.cpp）を指定すると、フィルタの条件を満たすファイルは最上位のプロジェクトを作成せずに直接追加されます。ただし、サブディレクトリも追加される場合は、サブプロジェクトがそれらに対して作成され、フィルタの条件を満たすファイルのみが追加されます。

- b (ディレクトリが選択されている場合のみ) 選択したディレクトリのサブディレクトリにあるファイルもすべて追加する場合は、〈サブディレクトリの作業ファイルも含む〉チェックボックスを選択します。
- c 〈コメント〉フィールドに、追加する作業ファイルの説明を入力します。このフィールドには必ず入力しなければなりません。ここに説明を入力するまでは、【OK】は無効です。
- d (複数の作業ファイルまたはディレクトリ全体を追加する場合のみ) 各ファイルの一意なコメントの入力を求めるようにするには、〈このコメントをすべてに適用〉チェックボックスの選

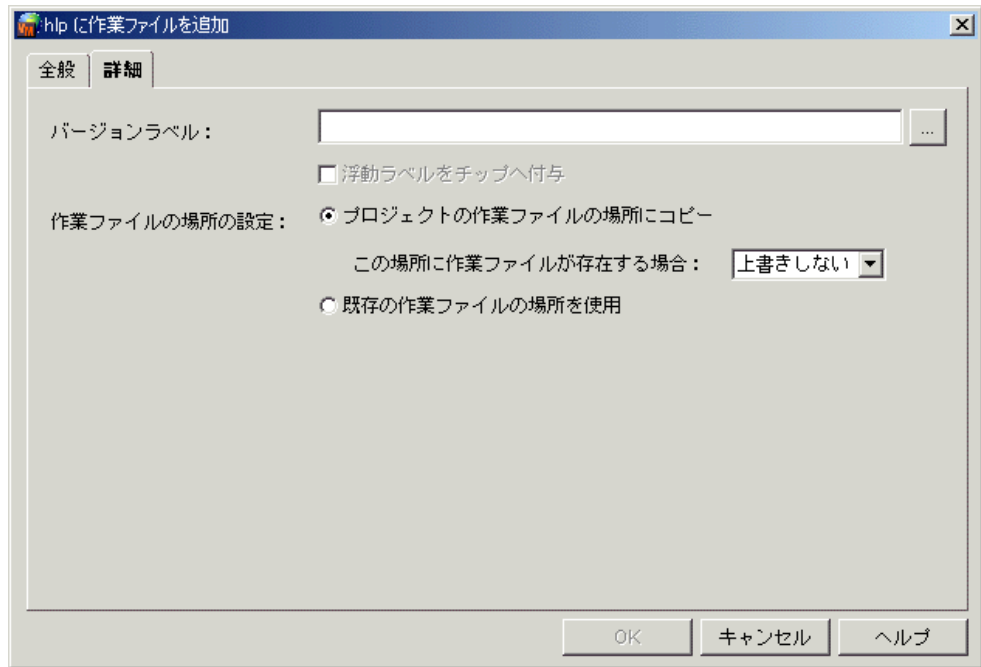
択を解除します。解除しないと、同じコメントが各ファイルに対して使用されるようになります。

- e** 〈バージョン管理ファイルが存在する場合〉ドロップダウンリストで、プロジェクトに同名のバージョン管理ファイルがすでに含まれる場合の動作を選択します。
- 「上書きしない」を選択すると、プロジェクト内に存在する同名のバージョン管理ファイルが上書きされず、警告も表示されません。
 - 「上書きせずに警告を表示」を選択すると、プロジェクト内に存在する同名のバージョン管理ファイルが上書きされず、プロジェクト内に同名のファイルが存在していることを告げる警告が表示されます。これがデフォルトの設定です。
- f** 〈チェックイン後の作業ファイル〉ドロップダウンリストで、作業ファイルをチェックインした後の動作を選択します。
- 「書き込み禁止で残す」を選択すると、作業ファイルの場所に読み取り専用の作業ファイルが保持されます。これがデフォルトの設定です。
 - 「チェックアウト状態を保持」を選択すると、チェックイン後の新しいリビジョンがロックされます。
 - 「削除する」を選択すると、チェックイン後に、作業ファイルが作業ファイルの場所から削除されます。
- g** (プロモーションモデルが作成されている場合のみ) 〈プロモーショングループ〉ドロップダウンリストで、以下のいずれかを選択します。
- 追加する作業ファイルの最初のリビジョンに割り当てる最下位のプロモーショングループ
 - リビジョンにプロモーショングループを割り当てない場合は、「なし」
- デフォルト値は「デフォルトプロモーショングループ」です。これは、このアクションに使用する最下位のプロモーショングループを定義するワークスペース設定です。このワークスペース設定の値が定義されておらず、ここでも値を選択しない場合は、リビジョンにはプロモーショングループが割り当てられません。
- h** (SuperUser または Unlimited 権限を持っている場合のみ) 作業ファイルを追加 (アーカイブを作成) するが、バージョン管理ファイルの最初のリビジョンをチェックインしない場合は、〈チェックインしない〉チェックボックスを選択します。
- i** (TrackerLink および SourceBridge ユーザのみ) 追加する作業ファイルと Issue を関連付ける場合は、【Issue の関連付け】をクリックします。[TrackerLink または SourceBridge の関連付け] ダイアログボックスが表示されます。



注 Version Manager が、作業ファイルを追加するときに Issue との関連付けを要求するよう設定されている場合は、TrackerLink または SourceBridge が自動的に起動します。120 ページの『TrackerLink と SourceBridge のセットアップ』を参照してください。

- 4 作業ファイルをチェックインする場合は、必要に応じて、[詳細] タブで以下の操作を行うことができます。



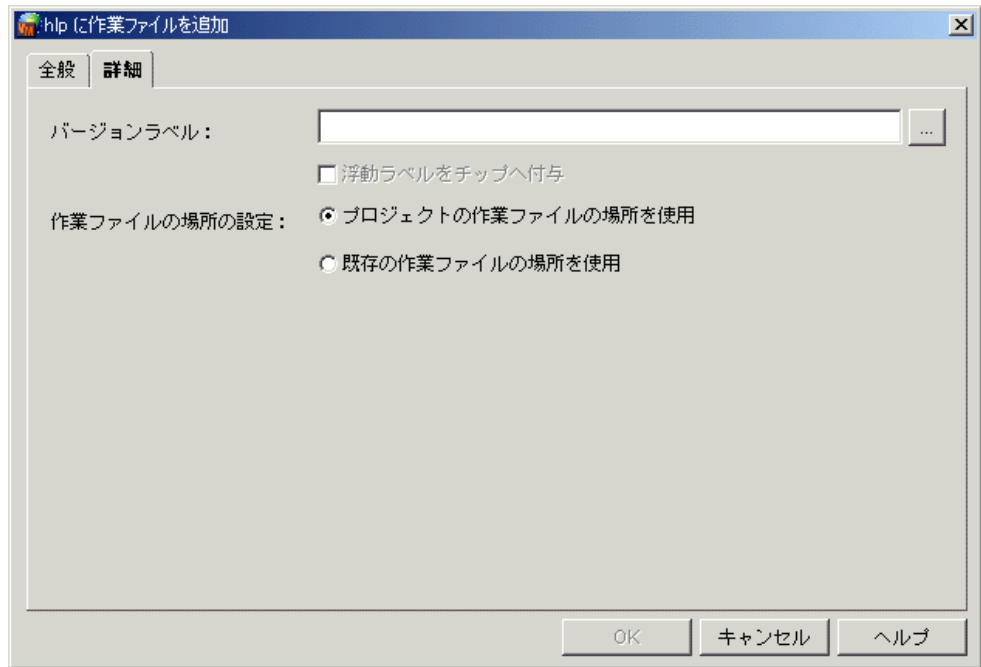
- a <バージョンラベル> フィールドに、チェックインするリビジョンのバージョンラベルを入力します。または、[...] をクリックして、バージョン管理ファイルに割り当てられている既存のバージョンラベルを選択します。バージョンラベルでは、大文字と小文字が区別されます。
- b バージョンラベルを常に最新のリビジョンに割り当てておくには、<浮動ラベル> チェックボックスを選択します。
- c 以下のいずれかを行って、追加する作業ファイルの場所を選択します（以下のオプションは、[全般] タブで <チェックインしない> チェックボックスが選択されているか、作業ファイルがすでに作業ファイルの場所にある場合は設定できません）。
 - ファイルを現在の場所からプロジェクトの作業ファイルの場所にコピーするには、<プロジェクトの作業ファイルの場所にコピー> ラジオボタンを選択します。プロジェクトの作業ファイルの場所が、作業ファイルの場所として使用されます。

このオプションを選択すると、作業ファイルの場所に同名の作業ファイルがあった場合の動作（「上書きしない」または「上書きする」）を選択することができます。

 - 作業ファイルの現在の場所を、作業ファイルの場所として使用するには、<既存の作業ファイルの場所を使用> ラジオボタンを選択します。



注 作業ファイルの場所以外から作業ファイルを追加している場合は、上記の設定のみが選択可能です。作業ファイルがすでに作業ファイルの場所にある場合は、ラジオボタンの選択を変更することはできず、以下の図のように <プロジェクトの作業ファイルの場所を使用> ラジオボタンが自動的に選択されます。



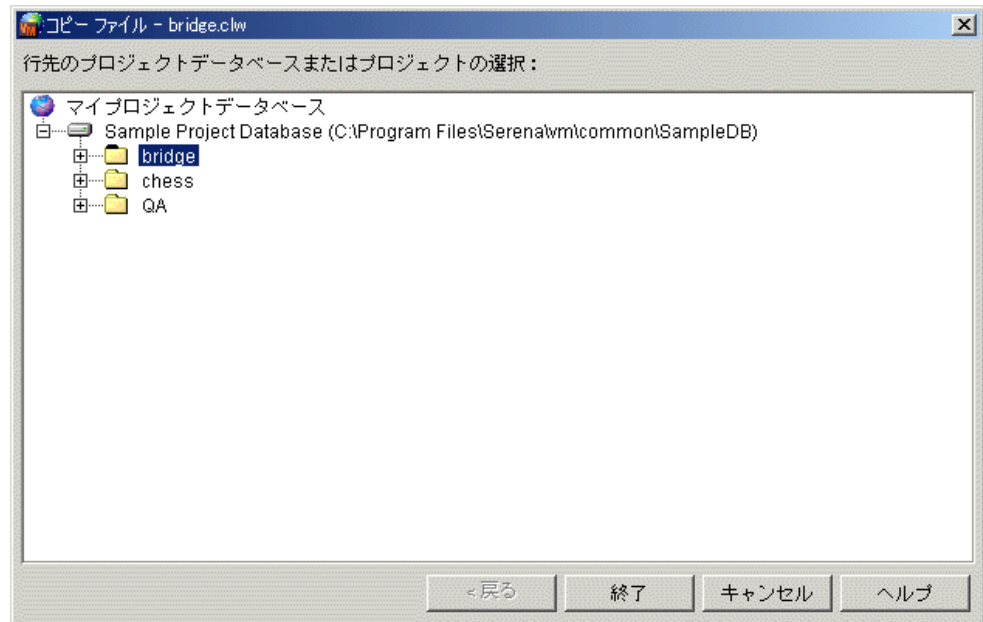
- 5 【OK】をクリックします。

アーカイブの共有

複数のプロジェクトによって使用されるファイルが存在する場合は、それらのファイルのアーカイブをほかのプロジェクトと共有できます。共有ファイルについては、プロジェクトごとにアーカイブを作成する必要はありません。その代わりに、バージョン管理ファイルをほかのプロジェクトにコピーし、既存の場所にあるアーカイブを使用します。コピーされたバージョン管理ファイルは、元のアーカイブを参照します。

アーカイブを共有するには

- 1 共有（コピー）するバージョン管理ファイルを選択します。
- 2 [編集] から [コピー] を選択します。[コピーファイル] ダイアログボックスが表示されます。



- 3 対象となるプロジェクトを選択します。このプロジェクトは、手順 1 で選択したバージョン管理ファイルのアーカイブを共有します。
- 4 【終了】をクリックします。

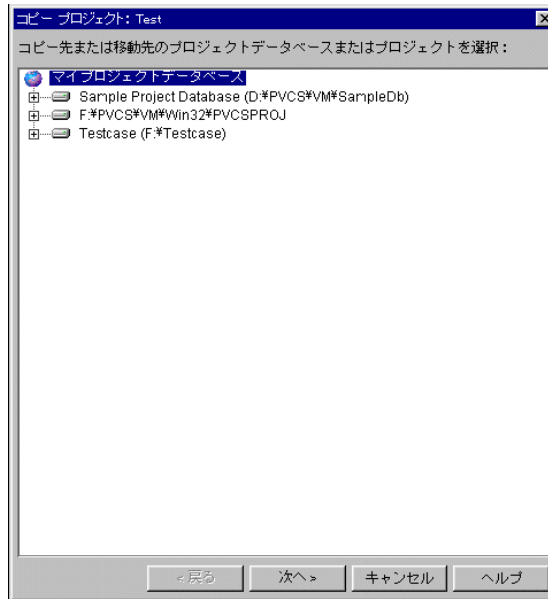
5.3/6.0 プロジェクトのプロジェクトデータベースへのコピー

5.3/6.0 プロジェクトをアップグレードするには、5.3/6.0 プロジェクトをプロジェクトデータベースにコピーします。また、5.3/6.0 プロジェクトルートをコピーして、新しいプロジェクトデータベースにアップグレードすることもできます。5.3/6.0 プロジェクトルートをコピーする前にプロジェクトデータベースを作成しておく必要はありません。

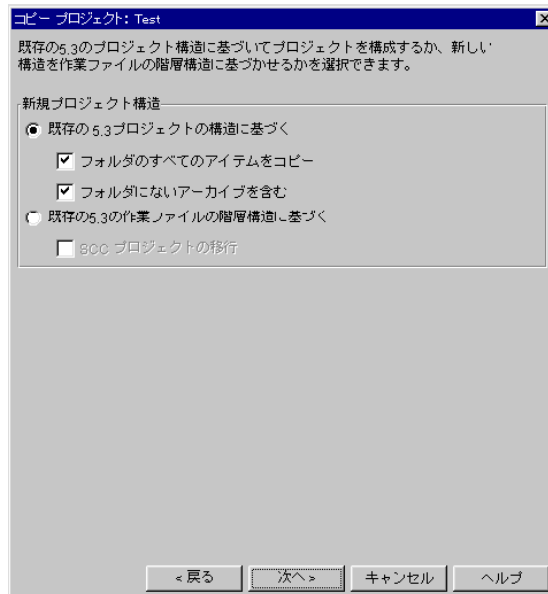
また、Version Manager Development Interface を使用して作成された 5.3/6.0 プロジェクトもアップグレードすることができます。これらのプロジェクトのアップグレードについて詳しくは、『Serena ChangeMan Version Manager Development Interface 操作ガイド』を参照してください。

5.3/6.0 プロジェクトをコピーするには

- 1 コピー（アップグレード）する 5.3/6.0 プロジェクトを選択します。
- 2 [編集] - [コピー] を選択します。[コピープロジェクト] ダイアログボックスが表示されます。

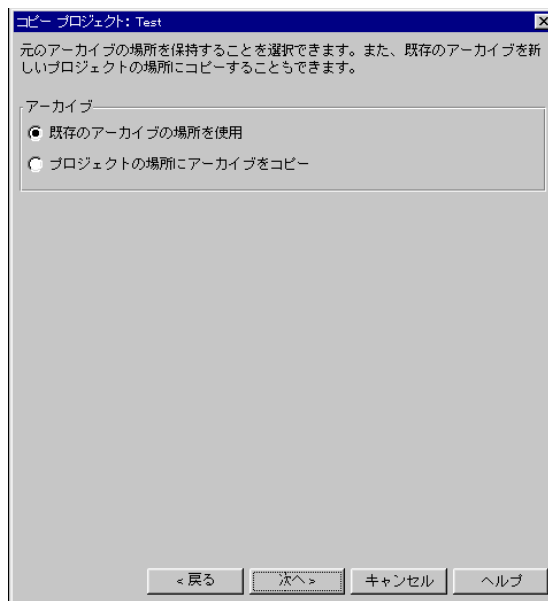


- 3 プロジェクトのコピー先のプロジェクトデータベース、またはプロジェクトデータベース内のプロジェクトを選択し、【次へ】をクリックします。2 番目の〔コピープロジェクト〕ダイアログボックスが表示されます。



- 4 次のいずれかを選択して、新しいプロジェクトの構造を決定します。
 - **既存の 5.3/6.0 プロジェクト構造に基づく**：新規プロジェクトを既存の 5.3/6.0 プロジェクトの構造を基に構成するには、このラジオボタンを選択します。この構造には、プロジェクトとサブプロジェクトという 2 つのレベルのみが含まれます。デフォルトでは、このオプションが選択されています。
 - アーカイブの場所にあるフォルダを含むすべてのアーカイブをコピーする場合は、〈フォルダ内の全項目をコピー〉チェックボックスを選択します。これはデフォルトで選択されています。

- フォルダに含まれないアーカイブ（5.3/6.0 プロジェクトのルートにあるアーカイブ）をコピーする場合は、〈フォルダにないアーカイブも含む〉チェックボックスを選択します。これはデフォルトで選択されています。
 - **既存の 5.3/6.0 作業ファイル階層構造に基づく**：作業ファイルの階層構造と同じ構造のプロジェクトを作成する場合は、このラジオボタンを選択します。この構造には、作業ファイルの場所と完全に一致する数のサブプロジェクトが含まれます。
 - **SCC 5.3/6.0 プロジェクトをアップグレードする場合は、〈SCC プロジェクトの移行〉**チェックボックスを選択します。このためには、〈既存の 5.3/6.0 作業ファイル階層構造に基づく〉ラジオボタンが選択されている必要があります。
- 5 【次へ】をクリックします。3 番目の〔コピープロジェクト〕ダイアログボックスが表示されず。

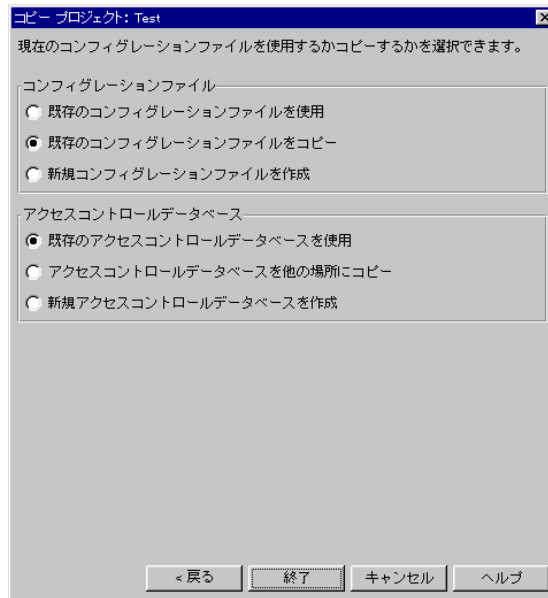


- 6 〈アーカイブ〉グループから、次のいずれかを選択します。
- 〈既存のアーカイブの場所を使用〉を選択すると、引き続き、現在の場所にある既存のアーカイブを参照します。つまり、コピーされたプロジェクトは 5.3/6.0 プロジェクトとアーカイブを共有します。これが、デフォルトの設定です。
- このプロジェクトについては、プロジェクトデータベースの archives ディレクトリの下に、新しいアーカイブディレクトリが作成されます（たとえば prjdb¥archives¥copiedprj）。コピーされたプロジェクトに新しく追加されたファイルは、5.3/6.0 プロジェクトのアーカイブの場所ではなく、プロジェクトデータベースのアーカイブ場所を参照します。
- 〈プロジェクトの場所にアーカイブをコピー〉を選択すると、アーカイブのコピーが作成され、新規プロジェクトの下の新しい archives ディレクトリに置かれます。オリジナルのアーカイブは現在の場所に置かれたままです。



注（SCC プロジェクトを移行している場合のみ）変換しているプロジェクトが、ほかの Development Interface プロジェクトと共有されている場合は、アーカイブを新しいプロジェクトディレクトリに移動しないことをお勧めします。アーカイブを移動すると、アーカイブとほかのプロジェクトとの関連付けが失われます。

- 7 【次へ】をクリックします。4 番目の〔コピープロジェクト〕ダイアログボックスが表示されず。

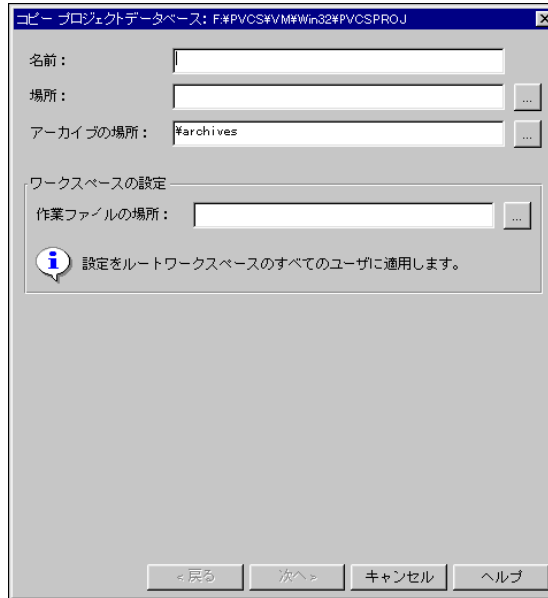


- 8 対象とするプロジェクトがコンフィグレーションファイルを使用しない場合には、〈コンフィグレーションファイル〉グループ内のオプションはグレー表示になります。コンフィグレーションファイルが使用されている場合には、次のいずれかを選択します。
- 〈既存のコンフィグレーションファイルを使用〉を選択すると、既存のコンフィグレーションファイルが、引き続き参照されます。したがって、5.3/6.0 プロジェクトと新しいプロジェクトにより、既存のコンフィグレーションファイルが共有されます。いずれかのプロジェクトがアクセスコントロールデータベースを使用している場合は、既存のコンフィグレーションファイルを使用しないことをお勧めします。
このオプションを選択すると、〈アクセスコントロールデータベース〉グループボックスのオプションはグレー表示となります。
 - 〈既存のコンフィグレーションファイルをコピー〉を選択すると、既存のコンフィグレーションファイルのコピーが作成され、新しいプロジェクトの構造中に配置されます。これがデフォルトの設定です。
 - 〈新規コンフィグレーションファイルを作成〉を選択すると、プロジェクトに対して新しいコンフィグレーションファイルが作成され、これには設定は定義されていません。
- 9 〈コンフィグレーションファイル〉グループで既存のコンフィグレーションファイルの使用を選択した場合は、〈アクセスコントロールデータベース〉オプションはグレー表示となります。それ以外の場合は、次のいずれかを選択できます。
- 〈既存のアクセスコントロールデータベースを使用〉ラジオボタンを選択すると、既存のアクセスコントロールデータベースが引き続き参照されます。これがデフォルトの設定です。
 - 〈アクセスコントロールを他の場所にコピー〉ラジオボタンを選択すると、既存のすべてのアクセスコントロールデータベースのコピーが作成され、新しいプロジェクトの構造中に配置されます。
 - 〈新規アクセスコントロールデータベースを作成〉を選択すると、新しいアクセスコントロールデータベースが作成されます。デフォルトでは、この新しいアクセスコントロールデータベースには操作を行っている管理者の ID によるユーザが 1 名含まれ、SuperUser 権限が割り当てられます。

10 【終了】をクリックします。

5.3/6.0 プロジェクトルートをコピーするには

- 1 コピーする 5.3/6.0 プロジェクトルートを選択します。
- 2 [編集] - [コピー] を選択します。[コピープロジェクトデータベース] ダイアログボックスが表示されます。



- 3 〈名前〉フィールドに新しいプロジェクトデータベースの名前を入力します。名前には、最初と最後にタブおよびスペースを使用できないこと以外に、使用できる文字の制限はありません。
- 4 〈場所〉フィールドで、プロジェクトデータベースの場所を指定します。この場所は、このプロジェクトデータベースにアクセスするすべてのユーザからアクセス可能な場所であることが必要です。

プロジェクトデータベースは、ほかのプロジェクトデータベースの下に作成することはできないため、ドライブのルートレベルには作成しないでください。また、既存の 5.3/6.0 プロジェクトルートと同じ場所にプロジェクトデータベースを作成することもできません。

- 5 アーカイブをデフォルト設定以外の場所に置く場合は、場所を〈アーカイブの場所〉フィールドで指定します。デフォルトの場所は、プロジェクトデータベースの下の archives という名称のディレクトリです。この場所は、このプロジェクトデータベースにアクセスするすべてのユーザからアクセス可能な場所であることが必要です。
- 6 〈作業ファイルの場所〉フィールドに作業ファイルの場所を指定します。このフィールドは必ず入力する必要があります。入力しなければ、次の手順に進めません。

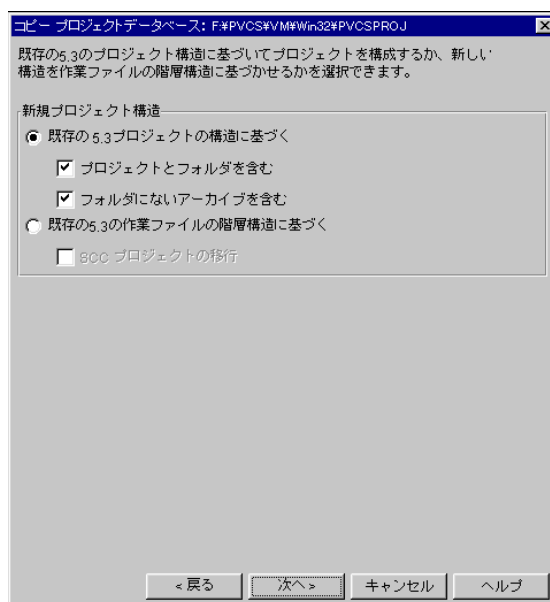
UNIX では、作業ファイルの場所にはパスのルートとして \$HOME を含められます（たとえば、\$HOME/work は /usr/cherylc/work に展開できます）。Version Manager は、作業ファイルの場所を判断するときに、HOME 環境変数の値を置き換えます。

\$HOME を使用することにより、パスは自動的にユーザの HOME 変数の値に応じて個別に設定されます。ユーザの HOME 変数の値が定義されていない場合は、作業ファイルの場所を判断するときにスペースで置き換えられます。



注 ほとんどのユーザにアクセス可能な、ネットワーク上のドライブを指定することを推奨します。ユーザは必要に応じて、作業ファイルの場所を変更するためのプライベートなワークスペースをローカルドライブに作成することができます。

- 7 【次へ】をクリックします。2 番目の [コピープロジェクトデータベース] ダイアログボックスが表示されます。

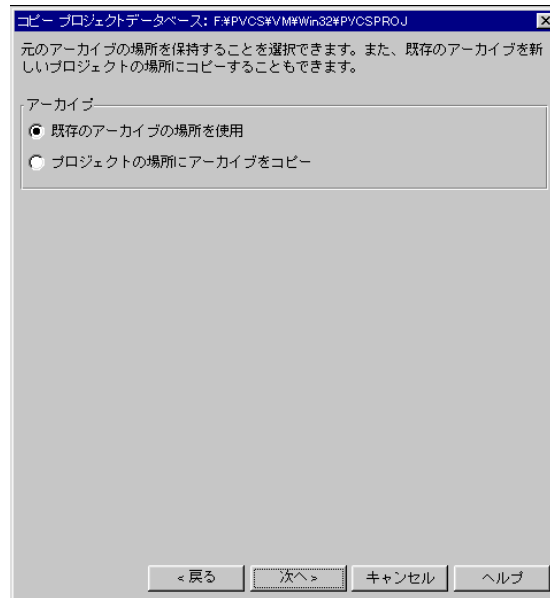


- 8 次のいずれかを選択して新しいプロジェクトデータベースの構造を指定します。
- 〈既存の 5.3/6.0 プロジェクト構造に基づく〉ラジオボタンを選択すると、既存の 5.3/6.0 プロジェクトルートに基づいて、新しいプロジェクトデータベースが作成されます。このプロジェクト構造には、プロジェクトとサブプロジェクトの 2 つまでのレベルが含まれます。これがデフォルトの設定です。

この構造は後日、プロジェクトをほかのプロジェクトに移動して変更することができます。プロジェクトをほかのプロジェクトに移動しても、移動されたプロジェクトは、同じアーカイブおよび作業ファイルの場所を参照します。

 - アーカイブの場所にあるフォルダを含むすべてのアーカイブをコピーする場合は、〈プロジェクトとフォルダを含む〉チェックボックスを選択します。これはデフォルトで選択されています。
 - フォルダに含まれないアーカイブ（5.3/6.0 プロジェクトのルートにあるアーカイブ）をコピーする場合は、〈フォルダにないアーカイブも含む〉チェックボックスを選択します。これはデフォルトで選択されています。
 - 〈既存の 5.3/6.0 作業ファイル階層構造に基づく〉を選択すると、新しいプロジェクトは、5.3/6.0 プロジェクトの作業ファイル階層構造に基づくものになります。このオプションの利点は、作業ファイルの整理にサブディレクトリを使用していた場合に、ネストされたプロジェクト構造が作成されることです。
 - 〈SCC プロジェクトの移行〉チェックボックスを選択すると、SCC 5.3/6.0 プロジェクトがアップグレードされます。このためには、〈既存の 5.3/6.0 作業ファイル階層構造に基づく〉ラジオボタンが選択されている必要があります。

- 9 【次へ】をクリックします。3 番目の [コピープロジェクトデータベース] ダイアログボックスが表示されます。



- 10 〈アーカイブ〉グループから、次のいずれかを選択します。

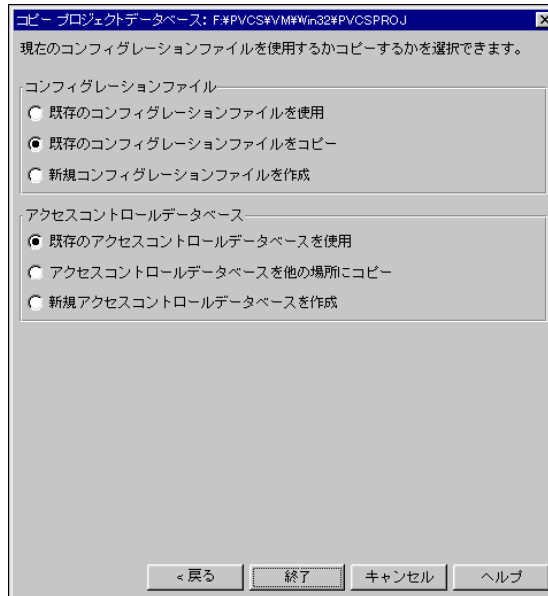
- 〈既存のアーカイブの場所を使用〉を選択すると、現在の場所にある既存のアーカイブを引き続き参照します。これがデフォルトの設定です。

プロジェクトデータベースに対しては archives ディレクトリが作成され、プロジェクトデータベースの各プロジェクトごとに個別にディレクトリが設定されます (たとえば prjdb¥archives¥projecta、prjdb¥archives¥projectb など)。

コピーされたプロジェクトに新しく追加されたファイルは、5.3/6.0 プロジェクトのアーカイブの場所ではなく、プロジェクトデータベースのアーカイブ場所を参照します。

- 〈プロジェクトの場所にアーカイブをコピー〉を選択すると、アーカイブのコピーが作成され、新規プロジェクトの下の新しい archives ディレクトリに置かれます。

- 11 【次へ】をクリックします。4 番目の〔コピープロジェクト〕ダイアログボックスが表示されます。



- 12 対象とするプロジェクトがコンフィグレーションファイルを使用しない場合は、〈コンフィグレーションファイル〉グループボックスのオプションはグレー表示になります。コンフィグレーションファイルが使用されている場合は、次のいずれかを選択します。

- 〈既存のコンフィグレーションファイルを使用〉ラジオボタンを選択すると、既存のコンフィグレーションファイルが引き続き参照されます。したがって、5.3/6.0 プロジェクトルートと新しいプロジェクトデータベースにより、既存のコンフィグレーションファイルが共有されます。プロジェクトルートまたはその下のいずれかのプロジェクトがアクセスコントロールデータベースを使用している場合は、既存のコンフィグレーションファイルを使用しないことをお勧めします。

このオプションを選択すると、〈アクセスコントロールデータベース〉グループボックスのオプションはグレー表示となります。

- 〈既存のコンフィグレーションファイルをコピー〉ラジオボタンを選択すると、既存のコンフィグレーションファイルのコピーが作成され、新しいプロジェクトの構造中に配置されます。これがデフォルトの設定です。
- 〈新規コンフィグレーションファイルを作成〉ラジオボタンを選択すると、プロジェクトに対して新しいマスターコンフィグレーションファイルが作成されます。

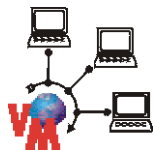
- 13 〈コンフィグレーションファイル〉グループボックスで既存のコンフィグレーションファイルの使用を選択した場合は、〈アクセスコントロールデータベース〉グループボックスのオプションはグレー表示となります。それ以外の場合は、次のいずれかを選択できます。

- 〈既存のアクセスコントロールデータベースを使用〉ラジオボタンを選択すると、既存のアクセスコントロールデータベースが引き続き参照されます。これがデフォルトの設定です。
- 〈アクセスコントロールを他の場所にコピー〉ラジオボタンを選択すると、既存のすべてのアクセスコントロールデータベースのコピーが作成され、新しいプロジェクトデータベースの構造中に配置されます。
- 〈新規アクセスコントロールデータベースを作成〉ラジオボタンを選択すると、新しいアクセスコントロールデータベースが作成されます。デフォルトでは、この新しいアクセスコントロールデータベースには操作を行っている管理者の ID によるユーザが 1 名含まれ、SuperUser 権限が割り当てられます。

- 14 【完了】をクリックします。

クロスプラットフォーム環境での Version Manager のセットアップ

このセクションでは、Version Manager を設定して、UNIX と Windows との間でアーカイブ、コンフィグレーションファイル、アクセスコントロールデータベース、およびプロジェクトデータベースを共有し、ディレクトリとファイルに権限を割り当てる手順について説明します。



注 アーカイブに対するすべてのアクセスに Version Manager File Server を使用している場合は、このセクションの説明は一部のみが適用されます。該当箇所の左には、それを示すグラフィックが付いています。Version Manager File Server の詳細については、[129 ページの『Version Manager File Server の設定と使用』](#)を参照してください。

このセクションの手順では、次の 2 つを前提としています。

- Version Manager が Windows と UNIX の両方のシステムにインストールされていること。
- UNIX と Windows の間でファイルシステムを共有するために、NFS または Samba が使用されていること。



注 Windows から Version Manager を実行しているが、アーカイブは UNIX に保存している場合は、Windows NFS トランスレーションのために、Windows ユーザのすべてのユーザ ID が同一の UNIX プライマリグループに属している必要があります。

Version Manager を実装するために作成するコンフィグレーションファイルには、Windows パスも含める必要があります。Windows ドライブ番号または UNC パスは、nfsmap ファイルにより、対応する UNIX パス名に対してマップされるため、同一のコンフィグレーションファイルを UNIX と Windows の両方に使用できます。

クロスプラットフォーム環境の UNIX における setuid の使用については、『Serena ChangeMan Version Manager インストールガイド』を参照してください。

次に、Version Manager をクロスプラットフォーム環境で使用するための、基本的な設定手順を示します。手順の詳細については以降のセクションで説明しています。

- 1 UNIX では、*pvcs* などの Serena 管理者ユーザ ID と、(まだ作成されていない場合は) すべての PVCS ユーザを含む *pvcsgrp* などのグループを作成することをお勧めします。このグループをプライマリグループとするか、各 PVCS ファイルとディレクトリのグループ所有権を PVCS グループに変更します。

Serena 管理者は、コンフィグレーションファイル、アクセスコントロールデータベース、nfsmap ファイル、および Version Manager 実行ファイルを含むすべてのディレクトリ、およびそれらのディレクトリ中のファイルのすべてを所有する必要があります。

ユーザ ID とグループの作成については、オペレーティングシステムのマニュアルを参照してください。

- 2 クロスプラットフォーム環境で Version Manager を使用するには、setuid モードを終了します。

- 3 Windows と UNIX システムの両方のユーザがコンフィグレーションファイルを共有できるように、UNIX 上で nfsmap ファイルを編集して Windows のドライブ番号をそれに対応する UNIX のパス名にマップします。



注

- この手順はアーカイブを作成する前に行う必要があります。
- nfsmap ファイルのエントリ数は 26 有効（コメントではない）エントリに制限されています。それ以上のエントリは無視されます。

- 4 UNIX と Windows で適切な権限を割り当てます。

- UNIX の場合：

ユーザ	割り当てる権限	対象
PVCS ユーザ および管理者	読み込み、書き込み、実行	コンフィグレーションファイルで指定されたディレクトリ
PVCS ユーザ	読み込み、実行	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ
PVCS ユーザ	読み込み	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル
<p>注 アクセスコントロールデータベースに自動的にユーザが作成されるようにし、ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。</p>		
PVCS 管理者	読み込み、書き込み、実行	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ
PVCS 管理者	読み込み、書き込み	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル

- Windows の場合：

ユーザ	割り当てる権限	対象
PVCS ユーザ および管理者	すべての権限	コンフィグレーションファイルで指定されたディレクトリ
PVCS ユーザ	読み込み	マスターコンフィグレーションファイルおよびアクセスコントロールデータベースを含むディレクトリとそのディレクトリ内のファイル
<p>注 アクセスコントロールデータベースに自動的にユーザが作成されるようにし、ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。</p>		
PVCS 管理者	すべての権限	マスターコンフィグレーションファイルおよびアクセスコントロールデータベースを含むディレクトリとそのディレクトリ内のファイル

- 5 Windows 上で、プラットフォーム間で共有するプロジェクトデータベースを作成します。プロジェクトデータベースの作成は、Windows 版 Version Manager デスクトップクライアントを使用して行う必要があります。UNIX 版 Version Manager デスクトップクライアントを使用してプロジェクトデータベースを作成すると、Windows ユーザからはアクセスできなくなります。
- 6 Version Manager を次のように設定します。
 - 新規作成および既存のアーカイブを書き込み可能にする。
 - テキストファイルが保存された、新規作成および既存のアーカイブの行末 (EOL) シーケンスを変換する。
 - データの破損を防ぐため、バイナリファイルが保存された新規作成および既存のアーカイブの行末 (EOL) シーケンスは変換しない。
 - Windows と UNIX の間でユーザ ID が同様に読まれるよう、ユーザ ID で大文字と小文字が区別されないように設定する。
- 7 Windows と UNIX の間で、共有されるファイルとディレクトリの名称で、大文字と小文字の区別を一致させます。

setuid の無効化

UNIX 上では、次のコマンドを使用して Version Manager 実行ファイルの setuid 権限を禁止します。

```
chmod u-s executable
```

ここで *executable* は次のいずれかです。

get	regen	vjournal
ident	vcompres	vlog
vmrg	put	vcs
vpromote	pvcsvmsuid	vdel
vsql	vdiff	

nfsmap ファイルの編集

Version Manager には、Windows と UNIX システムでユーザがコンフィグレーションファイルを共有できるよう、nfsmap ファイルが用意されています。nfsmap ファイルは、Windows のドライブ番号または UNC パス名を、それに対応する UNIX パス名にマップします。このファイルは UNIX 上で保存されます。このファイルが存在する場合は、Version Manager はコンフィグレーションファイル内のパス名の読み込みおよび書き込み時に MS-DOS パスフォーマットを使用します。



注 この手順は、アーカイブを作成する前に完了する必要があります。

UNIX 上で nfsmap ファイルを編集するには

- 1 nfsmap ファイルをテキストエディタで開きます。nfsmap ファイルのデフォルトの場所は、以下のとおりです。

```
Install_Location/vm/common/bin/OS/nfsmap
```

ここで、*os* はオペレーティングシステムの名前です。たとえば、Solaris の場合、パスは次のようになります。

```
Install_Location/vm/common/bin/solaris/nfsmap
```

- 2 各マッピングをそれぞれの行の 2 列を使用して入力します。左の列にはドライブ文字または UNC パス名を、右の列にはそれに対応する UNIX ディレクトリを入力します。UNC パス名にスペースが含まれる場合は、それを二重引用符 (") で囲む必要があります。

例 1 : M: ドライブを UNIX ディレクトリである /product/dev にマップするには、次のように入力します。

```
M /product/dev
```

例 2 : UNC パス名 "¥¥myserver¥vol2 abc" を UNIX ディレクトリである /product/dev にマップするには、次のように入力します。

```
"¥¥myserver¥vol2 abc" /product/dev
```



重要！ ドライブ文字、UNC パス、UNIX ディレクトリはそれぞれ一意である必要があります。同じドライブ文字、UNC パス、UNIX ディレクトリを別のマッピングで繰り返し使用することはできません。

- 3 変更を保存し、テキストエディタを終了します。

権限の割り当て

UNIX と Windows の両方において、ユーザに次のように権限を割り当てます。

- ユーザは、アーカイブの読み込みと書き込みが可能である、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル (nfsmap ファイルは UNIX 上のみ) については読み込みのみが可能。



注 アクセスコントロールデータベースに自動的にユーザが作成されるようにし、ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。

- 管理者は、アーカイブ、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル (nfsmap ファイルは UNIX 上のみ) の読み込みが可能。ユーザには Version Manager の設定を変更する可能性があるため、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルに対する書き込み権は割り当てない。

UNIX および Windows で権限を割り当てるには

- 1 UNIX 上で、PVCS ユーザと Serena 管理者に、コンフィグレーションファイル内で指定された以下のディレクトリ (および該当するサブディレクトリ) に対する読み込み、書き込み、および実行の権限を割り当てます。
 - アーカイブディレクトリ (VCSDir)
 - セマフォディレクトリ (SemaphoreDir)

- 内部使用の一時ファイルが含まれるディレクトリ (WorkDir)
- 一時使用のアーカイブファイルディレクトリ (ArchiveWork)
- 参照用ディレクトリ (ReferenceDir および VCSDir)
- ジャーナルファイルディレクトリ (Journal)

それぞれのディレクトリについて、次のコマンドを入力します。

```
chmod ug+rxw directory
```

- 2 UNIX 上で、PVCS ユーザに Version Manager のコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリに対する読み込みと実行の権限を割り当てます。また、Serena 管理者には、このディレクトリに対する読み込み、書き込み、および実行の権限を割り当てます。

それぞれのディレクトリについて、次のコマンドを入力します。

```
chmod ug+rx,u+w directory
```

- 3 UNIX 上で、PVCS ユーザに手順 2 に記載されたファイルについて読み込みの権限のみを割り当てます。また、Serena 管理者には、これらのファイルについて読み込みと書き込みの権限を割り当てます。

それぞれのディレクトリに移動し、それぞれのファイルについて次のコマンドを入力します。

```
chmod ug+r,u+w file_name
```

- 4 Windows 上で、PVCS ユーザと Serena 管理者に、コンフィグレーションファイル内で指定された以下のディレクトリ（および該当するサブディレクトリ）に対するすべての権限を割り当てます。

- アーカイブディレクトリ (VCSDir)
- セマフォディレクトリ (SemaphoreDir)
- 作業ディレクトリ (WorkDir)
- アーカイブ作業ディレクトリ (ArchiveWork)
- 参照用ディレクトリ (ReferenceDir および VCSDir)
- ジャーナルファイルディレクトリ (Journal)

- 5 Windows 上で、PVCS ユーザに Version Manager のコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリに対する読み込み権限のみを割り当てます。ユーザには、これらのファイルの変更は許可しません。

- 6 Windows 上で、手順 5 に記載されたファイルに対するすべての権限を Serena 管理者に割り当てます。管理者にはこれらのファイルの変更を許可します。

Version Manager の設定

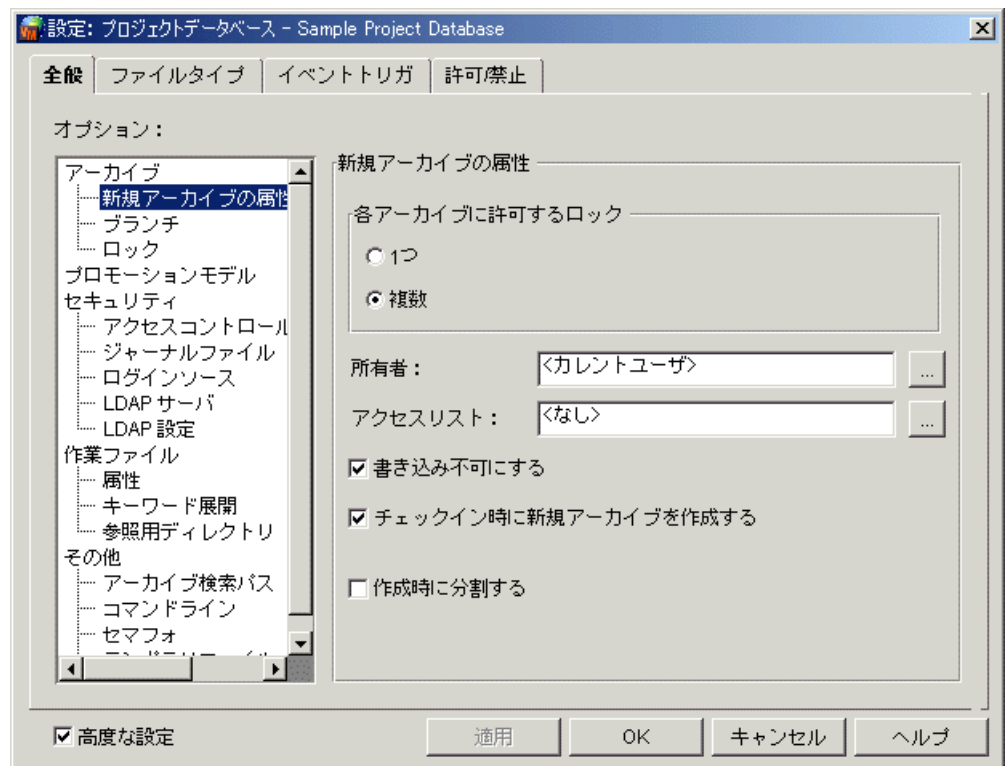
このセクションでは、Version Manager を次のように設定します。

- アーカイブを書き込み可能にする。
- テキストファイルの EOL シーケンスを変換し、バイナリファイルの EOL シーケンスは変換しない。
- ユーザ ID の大文字と小文字を区別しないようにする。

アーカイブへの書き込み許可

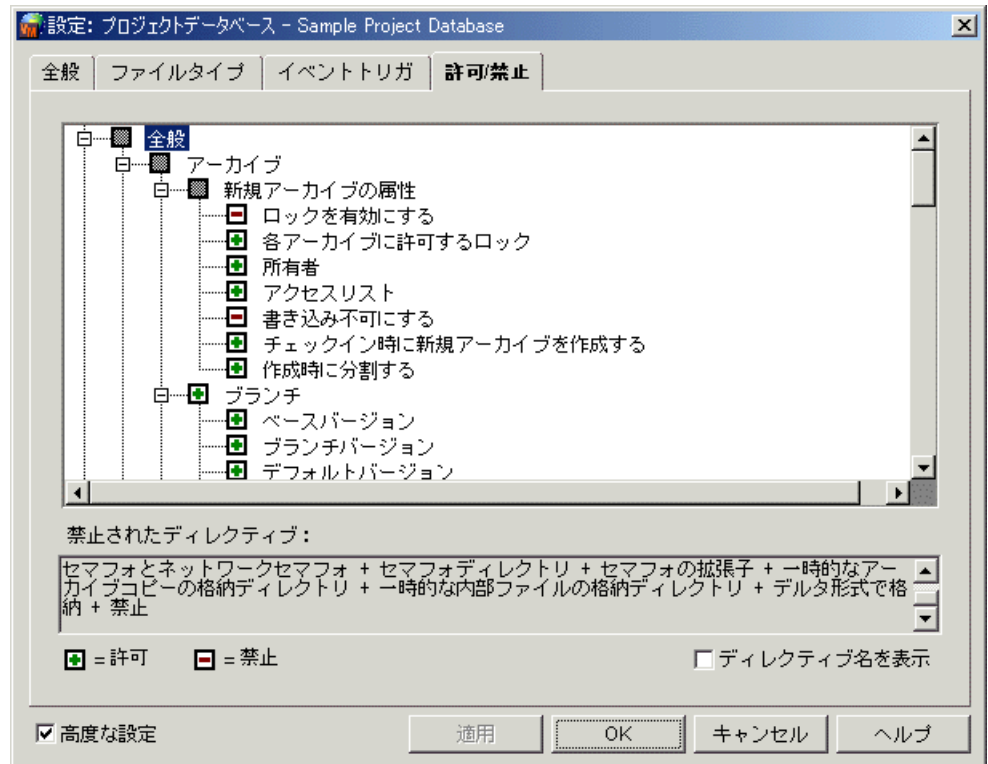
Windows と UNIX でアーカイブを共有する場合は、アーカイブファイルが書き込み可能となっている必要があります。Windows 上で、次の作業を行います。

- 1 次の作業により、新規作成のアーカイブが書き込み可能となるよう、各プロジェクトデータベースを設定します。
 - a 設定するプロジェクトデータベースを選択します。
 - b [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
 - c <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
 - d <オプション> リストボックスで、「アーカイブ」の下の「新規アーカイブの属性」を選択します。右側に <新規アーカイブの属性> 表示部が表示されます。



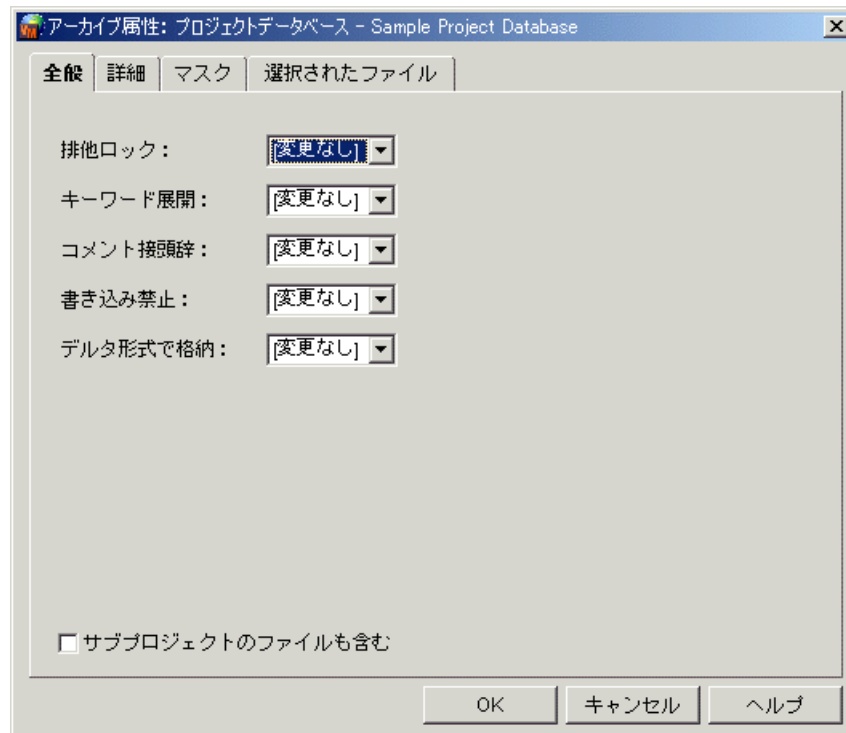
- e <書き込み不可にする> チェックボックスの選択を解除します。
- f 【適用】をクリックします。

- 2 ユーザが〈書き込み不可にする〉の選択を変更できないように、各プロジェクトデータベースを設定します。次の作業を行います。
 - a 【許可 / 禁止】タブを選択します。このタブには設定オプションのチェックボックスからなるツリーが表示され、それぞれについて許可または禁止を設定できます。



- b 〈書き込み不可にする〉オプションのチェックボックスにプラス記号 (+) が表示されている場合は、このチェックボックスをクリックしてマイナス記号 (-) にします。マイナス記号 (-) は、そのオプションが許可されていないことを示します。
 - c 【OK】をクリックします。
- 3 次の作業を実行して、プロジェクトデータベース中の既存の Version Manager アーカイブを、すべて書き込み可能とします。
 - a 書き込み可能にする既存のアーカイブのあるプロジェクトデータベースを選択します。

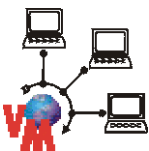
- b [管理] - [アーカイブ属性] を選択します。[アーカイブ属性] ダイアログボックスが表示され、[全般] タブがアクティブになっています。



- c <書き込み禁止> ドロップダウンリストで「変更なし」を選択します。
- d <サブプロジェクトのファイルも含む> チェックボックスを選択します。プロジェクトデータベース中のすべてのプロジェクト / サブプロジェクトについて、既存のアーカイブの属性が変更されるようになります。
- e 【OK】 をクリックします。

- 4 プラットフォーム間で共有するプロジェクトデータベースごとに、手順 1 から 3 までを繰り返します。

EOL シーケンスの変換



UNIX と Windows でアーカイブを共有するには、テキストファイルのリビジョンをチェックインするとき、キャリッジリターン (CR) である UNIX 上の EOL (行末) シーケンスをキャリッジリターン+ラインフィード (CR+LF) に変換し、リビジョンをチェックアウトするときには逆に変換するよう、各プロジェクトデータベースを設定する必要があります。

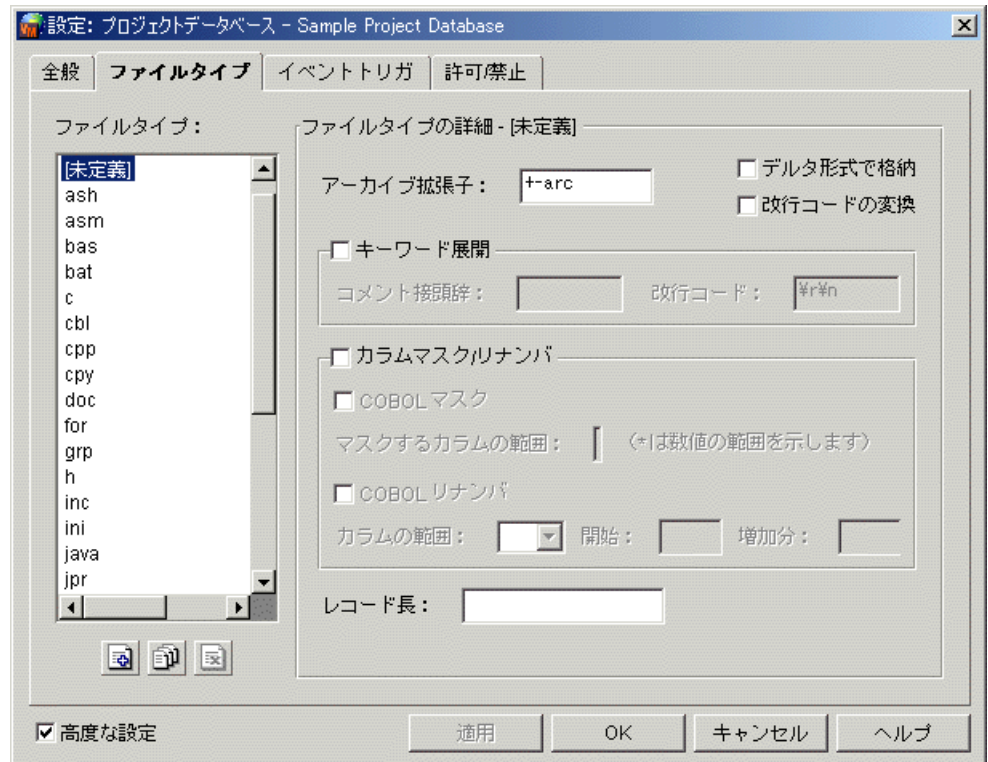


注意！ バイナリファイルの EOL シーケンスを変換すると、作業ファイルが破損することがあります。

テキストファイルの EOL シーケンスを変換するには

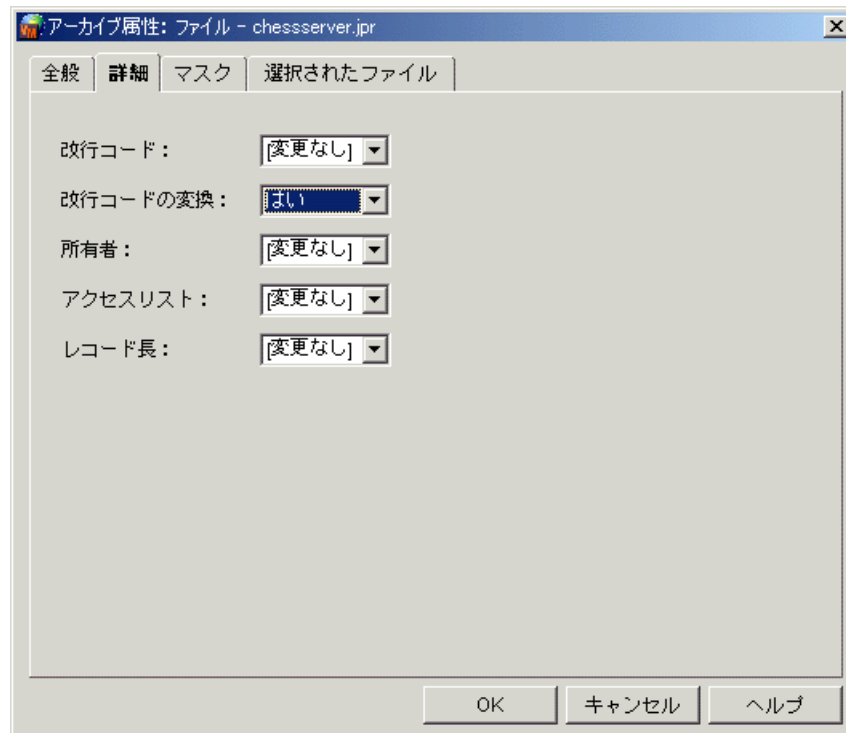
- 新規に作成されたアーカイブにおいて、テキストファイルのチェックインとチェックアウト時に EOL シーケンスを変換するよう、各プロジェクトデータベースを設定します。次の作業を行います。
 - 設定するプロジェクトデータベースを選択します。

- b [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- c [ファイルタイプ] タブを選択します。



- d 左側の〈ファイルタイプ〉リストから、EOL シーケンスを変換するファイルタイプ（テキストファイルの拡張子）を選択します。
 - e 〈改行コードの変換〉チェックボックスを選択します。
 - f テキストファイルの各拡張子について、手順 d と e を繰り返します。【OK】をクリックします。
- 2 次の作業を行い、プロジェクトデータベースの既存のすべてのアーカイブについて EOL シーケンスを変換するように設定します。
- a [表示] - [フィルタ] - [ワイルドカード] を選択して、テキストファイルの拡張子を持つバージョン管理ファイルをすべて表示します。たとえば、拡張子が .txt であるファイルをすべて表示します。
 - b 手順 a でフィルタされた、バージョン管理ファイルを選択します。
 - c [管理] - [アーカイブ属性] を選択します。[アーカイブ属性] ダイアログボックスが表示され、[全般] タブがアクティブになっています。

- d [詳細] タブを選択します。〈改行コードの変換〉ドロップダウンリストで「はい」を選択します。

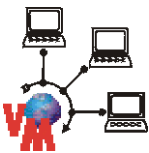


- e [OK] をクリックします。選択されたすべてのファイルについて、EOL シーケンスの変換が有効になります。

- f ほかのタイプのテキストファイルについて手順 2 を繰り返します。

- 3 デフォルトでは、バイナリファイルの EOL シーケンスは変換されません。したがって、バイナリファイルについてこのオプションを無効にする必要はありません。しかしバイナリファイルについてこのオプションが有効になっている疑いがある場合は、必ず無効にしてください。

ユーザ ID での大文字と小文字の区別

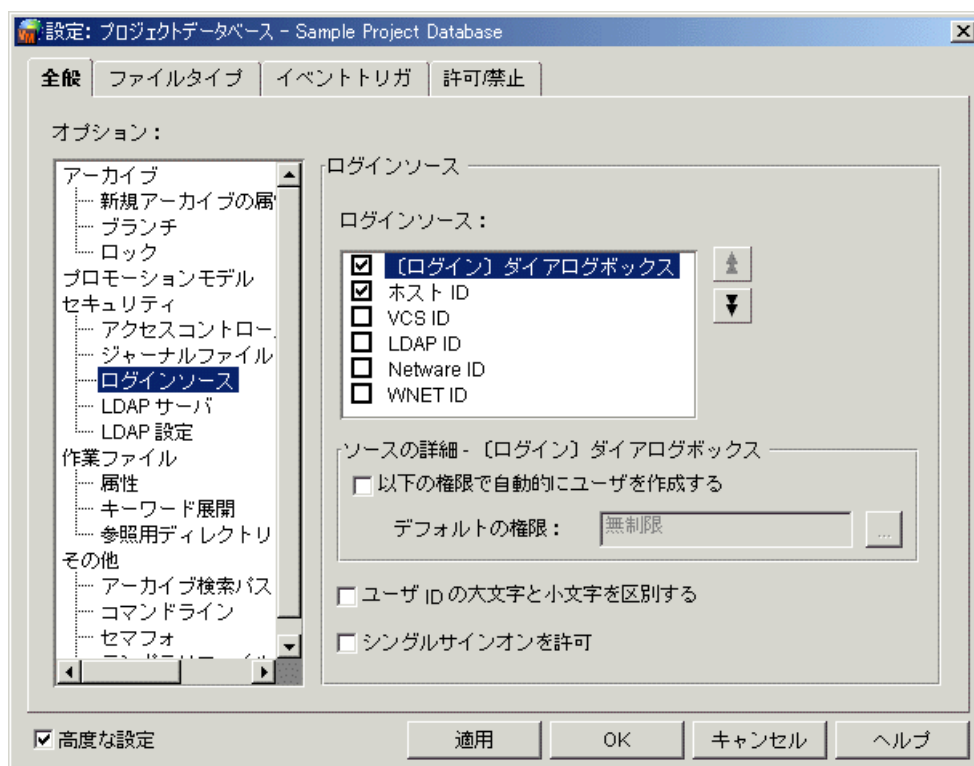


UNIX と Windows でユーザ ID が同様に読み込まれるようにするには、ユーザ ID の大文字と小文字が区別されないよう、各プロジェクトデータベースを設定する必要があります。

各プロジェクトデータベースでユーザ ID で大文字と小文字が区別されないよう設定するには

- 1 設定するプロジェクトデータベースを選択します。
- 2 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 3 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。

- 「セキュリティ」の下の「ログインソース」を選択します。[ログインソース] グループボックスが右側に表示されます。



- 〈ユーザ ID で大文字と小文字を区別〉チェックボックスの選択を解除します。
- 【OK】をクリックします。
- プラットフォーム間で共有するプロジェクトデータベースごとに、手順 1 から 6 までを繰り返します。

ファイル名およびディレクトリ名の大文字と小文字の使用



Windows と UNIX で共有されるファイルとディレクトリの名称において、大文字と小文字が一貫して使用されるよう、すべての名称について小文字を使用することをお勧めします。その場合、UNIX でも名称を読むことができます。

Windows で新規にアーカイブを作成する場合は、VCS コマンドを使用してアーカイブ中の作業ファイルの名称の大文字と小文字の使用を修正します。

次のステップ

Version Manager プロジェクトデータベースのセットアップが終わったら、次の作業を行うことができます。

- プロジェクトデータベースとプロジェクトの設定。69 ページの『Version Manager の設定』を参照してください。
- セキュリティの追加。201 ページの『セキュリティの使用』を参照してください。

-
- プロモーションモデルの定義。245 ページの『プロモーションモデルの使用』を参照してください。
 - イベントトリガの定義。271 ページの『イベントトリガの使用』を参照してください。

第 2 章

コマンドラインインターフェイスを使用した Version Manager のセットアップ

はじめに	53
Version Manager のコマンドラインインターフェイスについて	53
プロジェクト組織のシナリオ	55
Version Manager のセットアップ計画	56
ネットワークを使用した複数ユーザのプロジェクトのセットアップ	57
クロスプラットフォーム環境での Version Manager のセットアップ	60
次のステップ	67

はじめに

コマンドラインインターフェイスの使用により、Serena ChangeMan Version Manager を多様な管理スタイルやプロジェクト規模に使用できるようになります。この章ではまず最初に、コマンドラインインターフェイスの使用方法について説明します。次に、Version Manager のさまざまな設定方法について理解するため、いくつかのプロジェクトの例を示します。また、セットアップ計画、複数のユーザが関与するネットワークベースのプロジェクトのセットアップ手順、およびクロスプラットフォーム環境で使用するための Version Manager のセットアップ手順についても説明しています。

デスクトップクライアントを使用した Version Manager のセットアップについては、[15 ページの『デスクトップクライアントを使用した Version Manager の設定』](#)を参照してください。

Version Manager のコマンドラインインターフェイスについて

Version Manager 5.3/6.0 のコマンドラインインターフェイスと同様に、現在のバージョンの Version Manager コマンドラインインターフェイスを使用すると、アーカイブを直接操作することができます。

ただし、Version Manager デスクトップクライアントはプロジェクトデータベースを経由してアーカイブにアクセスするため、デスクトップクライアントでは使用できてもコマンドラインインターフェイスでは使用できないコマンドがいくつか存在します。さらに、コマンドラインインターフェイスとデスクトップクライアントのどちらを経由してアクセスするかにかかわらず、同一のデフォルトアーカイブ拡張子テンプレートが使用されることを保証するため、どちらのインターフェイスを使用する場合もアーカイブ更新時は同一のコンフィグレーションファイルを使用する必要があります。このセクションでは、これらの問題についても説明します。

デフォルトのアーカイブ拡張子テンプレート

Version Manager デスクトップクライアントとコマンドラインインターフェイスのどちらを経由してアクセスする場合にも、アーカイブには同一のアーカイブ拡張子テンプレートを使用する必要があります。アーカイブ拡張子テンプレートは、アーカイブの拡張子を決定するときに Version Manager が使用するパターンを規定します。Version Manager のデスクトップクライアントとコマンドラインインターフェイスでは、デフォルトのアーカイブ拡張子テンプレートは共通ではありません。

Version Manager インターフェイス	アーカイブ拡張子テンプレート	アーカイブ拡張子の例
デスクトップクライアント	+arc	readme.txt-arc
コマンドライン	??V____	readme.txv

デスクトップクライアント用テンプレートを使用すると、アーカイブの名前に基づいてファイル名を簡単に識別できるようになります。ただし、アーカイブにデスクトップクライアントとコマンドラインインターフェイスの両方からアクセスする場合は、両方のインターフェイスにおいて同じ拡張子テンプレートをアーカイブに適用する必要があります。

同じテンプレートが両方のインターフェイスに使用されていることを保証するためのもっとも簡単な方法は、同じコンフィグレーションファイルをコマンドラインインターフェイスとデスクトップクライアントの両方に使用することです。詳しくは、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

使用時のガイドライン

Version Manager デスクトップクライアントで作成されたアーカイブに、コマンドラインインターフェイスでアクセスするには、次のガイドラインに従ってください。

- Version Manager デスクトップクライアントで作成された同じコンフィグレーションファイルを常に使用する。
- Version Manager デスクトップクライアントには「取得」というコマンドが追加されており、これによりデフォルトのリビジョンの読み込み専用コピーが一時的にテンポラリディレクトリ (Windows 版では %temp%\pvcs、UNIX 版では /tmp/pvcs) に抽出されます。テンポラリディレクトリは、Windows 版では TEMP 環境変数、UNIX 版では pvcsvmux スクリプトにより定義されます。

コマンドラインインターフェイスからは、GET コマンドとコマンドオプションを使用して、デフォルトのリビジョンの最新バージョンの書き込み可能コピーをチェックアウトします。これら 2 つのコマンドは同様のものですが、インターフェイスによってその動作が若干異なります。

制限事項

Version Manager デスクトップクライアントで作成されたアーカイブにコマンドラインインターフェイスからアクセスする場合は、次の制限事項が適用されます。

- プロジェクトデータベースを作成する、開く、名称変更する、または閉じるなど、プロジェクトデータベースに関連する機能のコマンドは使用できない。コマンドラインは、直接アーカイブに対して動作します。
- プロジェクトを作成する、開く、名称変更する、または閉じるなど、プロジェクトおよびサブプロジェクトに関連する機能のコマンドは使用できない。コマンドラインは直接アーカイブに対して動作します。

- ワークスペースを作成する、開く、名称変更する、または閉じるなど、ワークスペースに関連する機能のコマンドは使用できない。コマンドラインインターフェイスでは、作業ファイルの場所は通常、現在のディレクトリになります。ファイルとともに別のパスを指定することにより、ほかの作業ファイルの場所も指定できます。次にその例を示します。

```
get foo.c-arc(e:¥temp)
```
- リストファイルも使用できる。これについては『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』に説明されています。

プロジェクト組織のシナリオ

このセクションでは、小規模な 1 名のみプロジェクトから、複数のスタッフが関わるネットワークベースのプロジェクトまで、プロジェクト組織のさまざまなシナリオを示します。

小規模な単一ユーザのプロジェクト

このタイプのプロジェクトでは、単一の開発者により最大 25 までのファイルが操作されます。Version Manager のデフォルト設定は、このサイズのプロジェクトに適しています。

すべてのソースおよびオブジェクトファイル、ビルドスクリプト、およびその他のファイルは、バックアップを簡単にするため、通常は 1 つのディレクトリに置かれます。アーカイブはソースファイルと同じディレクトリに置かれ、エラーに備えて書き込み保護が設定されます。これはフラットな（階層のない）構造です。

プログラマはファイルをチェックアウトしてロックをかけ、ファイルを更新したあと再びチェックインする必要があります。

大規模な単一ユーザのプロジェクト

このタイプのプロジェクトでは、1 名のプログラマが多数のファイルを保守します。関連するファイルは機能に応じてディレクトリにまとめられます。これは階層的な構造です。

各ディレクトリ内のファイル数を減らすため、1 つのディレクトリに関連ファイルをまとめて入れ、各ディレクトリにはファイルのアーカイブを格納するサブディレクトリを作成します。また、セキュリティのため、アーカイブには書き込み保護を行うこともお勧めします。

ネットワークを使用した複数ユーザのプロジェクト

開発者がネットワークを使用する場合は、プロジェクトチームの全員がプロジェクトファイルにアクセスできるよう、ファイルをネットドライブに配置できます。関連するファイルは、機能に応じてディレクトリにまとめられ、階層的な構造となります。

チームメンバーは通常、プロジェクトディレクトリの構造を自分のワークステーションに複製します。開発者が一度に更新するプロジェクトソースファイルは、一般にわずかな数に過ぎないため、ワークステーションには使用中のファイルのみが保存されます。

Version Manager のセットアップ計画

Version Manager のセットアップを行うには、あらかじめ次のような情報が必要となります。

- ソース管理を行う作業ファイル、またはソースファイルはどこにあるか。
- ファイルのアーカイブをどこに配置するか。
- ユーザにどこで作業ファイルを編集させるか。
- 使用環境がクロスプラットフォームかどうか。

作業ファイル

複数のユーザによる大規模プロジェクト、および大規模な単一ユーザのプロジェクトでは、ソース管理を行う作業ファイルが機能ごとにディレクトリにまとめられていれば、すでにかんりの作業が終了しています。ファイルは機能に従ってディレクトリにまとめ、この構造をアーカイブと作業ファイルの場所にも適用することをお勧めします。

小規模な単一ユーザのプロジェクトでは、すべてのファイルはアーカイブを含めて 1 つのディレクトリに入れられます。

アーカイブの場所

アーカイブの場所とは、作業ファイルに対するアーカイブを保管するためのディレクトリです。複数のユーザによる大規模プロジェクトでは、この場所は通常、許可を得たユーザがすべてアクセスできる、ネットワーク上の共有ディスクになります。また、アーカイブの場所の階層構造は、作業ファイルの場所の階層構造と同一にすることをおすすめします。



注 Version Manager File Server を使用している場合は、アーカイブはサーバ上にある可能性があります。129 ページの『Version Manager File Server の設定と使用』を参照してください。

作業ファイルの場所

作業ファイルの場所とは、アーカイブからチェックアウトされたリビジョンが、Version Manager によって格納される場所です。リビジョンをチェックアウトすると、作業ファイルになります。

複数のスタッフによる大規模プロジェクトでは、ユーザは自分のローカルのワークステーション上に、アーカイブ構造を反映した作業ファイルディレクトリを作成することをお勧めします。

クロスプラットフォーム環境

Version Manager のコマンドラインインターフェイスの使用により、Windows と UNIX で、コンフィグレーションファイル、アクセスコントロールデータベース、およびアーカイブを共有できます。

アーカイブを UNIX に保存して Windows からアクセスすることも、その逆も可能です。

ネットワークを使用した複数ユーザのプロジェクトのセットアップ

次に、単一プラットフォーム環境において、ネットワークを使用した複数のユーザによるプロジェクトをセットアップするための、基本的な手順を示します。手順の詳細については、以降のセクションで説明しています。



注 このセットアップが正しく機能するためには、すべてのユーザが同じ場所にインストールされた Version Manager を使用している必要があります。そのためには、全員がネットワークインストールを行うか、ネットワークインストールのワークステーションインストールを行う必要があります。

- 1 ネットワーク上に、プロジェクトのアーカイブディレクトリを作成します。ディレクトリの階層構造は、プロジェクトの構造を反映する必要があります。
- 2 ローカルのワークステーション上で、手順 1 で作成したアーカイブディレクトリ構造と同じになるようディレクトリを作成します。これらのディレクトリが作業ファイルの場所になります。管理者として作業ファイルを自分のワークステーション上のディレクトリに移動し、これらのディレクトリから作業ファイルをチェックインできるようにします。
- 3 手順 2 で作成した各作業ファイルディレクトリ内に、それらの作業ファイルに対するアーカイブディレクトリの場所を定義する、ローカルのコンフィグレーションファイル (vcs.cfg) を作成します。アーカイブディレクトリを定義するディレクティブは VCSDir です。
- 4 作業ファイルの最初のリビジョンをアーカイブディレクトリにチェックインし、各ファイルに対するアーカイブを作成します。
- 5 ネットワーク上にマスターコンフィグレーションファイルを作成し、このファイル内で WorkDir と ArchiveWork ディレクティブをネットワーク上でなくローカルの場所に設定します。こうすると、ネットワークトラフィックが減少してパフォーマンスが向上し、また複数のユーザが同じ名前のテンポラリファイルを作成して名前が衝突することも防止できます。
また、Semaphore ディレクティブを使用してアーカイブに対して同時に更新が行われることを防ぎ、SemaphoreDir ディレクティブを使用して共有セマフォディレクトリを設定します。
- 6 マスターコンフィグレーションファイルの場所をすべてのユーザに対して設定するため、VCONFIG コマンドを使用してマスターコンフィグレーションファイルの名称を Version Manager に埋め込みます。

アーカイブディレクトリの作成

- 1 プロジェクトの組織を反映する、階層構造を持つアーカイブディレクトリ構造を作成します。関連するファイルは、機能ごとにディレクトリにまとめる必要があります。
3 タイプのプロジェクト（たとえば include、resource、および source）がある場合は、MyApp などの名称のトップレベルのディレクトリを各ファイルタイプに対して作成します。



- 2 このアーカイブディレクトリに対する読み込み / 書き込み（および UNIX では実行）の権限が開発者に与えられていることを確認します。



注 アーカイブに対するすべてのアクセスに Version Manager File Server を使用している場合は、アーカイブディレクトリに対する読み取り / 書き込み許可または実行許可を有効にする必要はありません。129 ページの『Version Manager File Server の設定と使用』を参照してください。

ローカルワークステーションでの作業ファイルディレクトリの作成

この作業の手順 1 は、管理者として各開発者のワークステーションに対して行うか、各開発者に情報を提供して各自に行わせます。

- 1 ローカルのワークステーション上に、ネットワークのアーカイブディレクトリ構造と同じ構造を持つ作業ファイルディレクトリを作成します。
- 2 ワークステーションからソース管理を行うファイルをチェックインできるよう、ファイルを作業ファイルディレクトリに移動します。これらのファイルは、ローカルのコンフィグレーションファイルを作成した後にチェックインします。

ローカルコンフィグレーションファイルの作成

Version Manager がネットワークサーバ上のアーカイブを見つけられるよう、各作業ファイルディレクトリごとにローカルのコンフィグレーションファイル (`vcs.cfg`) を作成します。このファイルには、Version Manager がアーカイブを探すディレクトリを指定する `VCSDir` ディレクティブを含める必要があります。

- 1 テキストエディタを使用して新規ファイルを作成します。このファイルに次の行を入力します。

```
VCSDir = archive_path
```

ここで、`archive_path` は、アーカイブのディレクトリパスで、現在使用中のドライブと同じでない場合は、ドライブの記号 (Windows の場合) も含まれます (例: `f:¥myapps¥source`)。
- 2 ファイルを `vcs.cfg` という名前で、対応する作業ファイルディレクトリに保存します。
- 3 各作業ファイルディレクトリについて、手順 1 と 2 を繰り返します。
- 4 テキストエディタを終了します。

コンフィグレーションファイルの詳細な情報については、79 ページの『コマンドラインインターフェイスでのコンフィグレーションファイル』を参照してください。

作業ファイルのチェックイン

- 1 作業ファイルディレクトリに移動します。
- 2 各ファイルを、`vcs.cfg` ファイルで定義された適切なアーカイブディレクトリにチェックインします。次のコマンドを入力します。

```
put workfile_name
```

ここで、

`workfile_name` は、現在のディレクトリ中のファイルの名称です。現在のディレクトリからすべてのファイルをチェックインする場合には、「*」を指定してすべてのファイルを同時にチェックインできます。

たとえば、現在のディレクトリからすべてのファイルをチェックインするには、次のように入力します。

```
put *
```

また、現在のディレクトリから `retry.c` という名称のファイルをチェックインするには、次のように入力します。

```
put retry.c
```

マスターコンフィグレーションファイルの作成

マスターコンフィグレーションファイルには、Version Manager の全体的な動作を定義する、共通の設定オプションが含まれます。マスターコンフィグレーションファイルは、どのコンフィグレーションファイルよりも先に Version Manager に読み込まれるため、特定のディレクティブについてすべてのユーザが同一の設定を共有することが保証されます。ほかのコンフィグレーションファイル（たとえば `vcs.cfg` ファイル）がマスターコンフィグレーションファイル中の共通設定より優先されるのを防ぐには、`Disallow` ディレクティブを使用してユーザが設定をリセットできないようにします。詳細については、[79 ページの『コマンドラインインターフェイスでのコンフィグレーションファイル』](#)を参照してください。

ネットワークを使用した複数ユーザのプロジェクトでは、`WorkDir` と `ArchiveWork` ディレクティブをローカルの場所に設定します。こうすると、ネットワークトラフィックを減らしてパフォーマンスを向上させ、複数のユーザが同名のテンポラリファイルを作成して名前が衝突するのを防止できます。ユーザはローカルの環境に合わせて、ローカルのコンフィグレーションファイル（`vcs.cfg`）内の設定を変更できます。

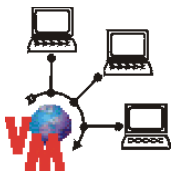
アーカイブに対して同時に更新が行われることを防ぐため、全員が同じタイプのセマフォを使用する必要があります。この場合は、`File` セマフォを使用します。また、セマフォディレクトリの定義も必要のため、`SemaphoreDir` ディレクティブを共有ディレクトリに設定します。次に、ユーザが `Semaphore` および `SemaphoreDir` ディレクティブを再定義できないよう、それらに対するアクセス権を禁止する必要があります。

マスターコンフィグレーションファイルには、これら以外にも多数の設定を行えます。コンフィグレーション設定とコンフィグレーションファイルの詳細については、[69 ページの『Version Manager の設定』](#)を参照してください。本製品には、デフォルトのマスターコンフィグレーションファイルが添付されています。このファイルは、インストールディレクトリの `lib` サブディレクトリ（たとえば、Windows では `C:\Program Files\Serena\vm\win32\lib`、UNIX では `/usr/pvcs/vm/solaris/lib`）にあり、名称は `default.cfg` です。

注 Version Manager ファイルサーバに作成したプロジェクトデータベースの場合、デフォルトのコンフィグレーションファイルは、`defaultfs.cfg` という名前になります。

マスターコンフィグレーションファイルに次の行を追加します。

```
WorkDir = path_name
ArchiveWork = path_name
Semaphore = File
Semaphore = network_path_name
Disallow Semaphore NoSemaphore
Disallow SemaphoreDir
```



ここで、*path_name* は任意のローカルディレクトリです。たとえば WorkDir については C:\¥PVCSTMP、ArchiveWork については C:\¥ARCTMP と定義できます。これらのディレクトリは、各ユーザのワークステーション上に存在しなければならないことに注意してください。

マスターコンフィグレーションファイル名の埋め込み

マスターコンフィグレーションファイルとして作成したコンフィグレーションファイルを Version Manager のコマンドラインインターフェイスで使用するには、このコンフィグレーションファイルの名称を Version Manager に埋め込む必要があります。



注 Version Manager に埋め込まれたマスターコンフィグレーションファイルは、そのファイルが埋め込まれた Version Manager のコピーを使用する、すべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。

ファイル名を埋め込むには、次のように **VCONFIG** コマンドを使用します。

```
vconfig -cconfig_filename vm_filename
```

ここで、

config_filename は、埋め込むコンフィグレーションファイルの名称です。

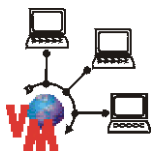
vm_filename は次のいずれかになります。

- Windows : VMWFVC.DLL
- UNIX : vmufvc.a

VCONFIG コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

クロスプラットフォーム環境での Version Manager のセットアップ

このセクションでは、Windows と UNIX でコンフィグレーションファイル、アクセスコントロールデータベース、およびアーカイブを共有するための Version Manager の設定、およびディレクトリとファイルへの権限の割り当てについて説明します。



注 アーカイブに対するすべてのアクセスに Version Manager File Server を使用している場合は、このセクションの説明は一部のみが適用されます。該当箇所の左には、それを示すグラフィックが付いています。Version Manager File Server の詳細については、[129 ページの『Version Manager File Server の設定と使用』](#)を参照してください。

このセクションの手順では、次の 2 つを前提としています。

- Version Manager が Windows と UNIX の両方のシステムにインストールされていること。

- UNIX と Windows の間でファイルシステムを共有するために、NFS または Samba が使用されていること。



注 Windows から Version Manager を実行しているが、アーカイブは UNIX に保存している場合は、Windows NFS トランスレーションのために、Windows ユーザのすべてのユーザ ID が同一の UNIX プライマリグループに属している必要があります。

Version Manager を実装するために作成するコンフィグレーションファイルには、Windows パスも含める必要があります。Windows ドライブ番号または UNC パスは、nfsmap ファイルにより、対応する UNIX パス名に対してマップされるため、同一のコンフィグレーションファイルを UNIX と Windows の両方に使用できます。

クロスプラットフォーム環境の UNIX における setuid の使用については、『Serena ChangeMan Version Manager インストールガイド』を参照してください。

次に、Version Manager をクロスプラットフォーム環境で使用するための、基本的な設定手順を示します。手順の詳細については以降のセクションで説明しています。

- 1 UNIX では、pvcs などの Serena 管理者ユーザ ID と、(まだ作成されていない場合は) すべての PVCS ユーザを含む pvcsgrp などのグループを作成することをお勧めします。このグループをプライマリグループとするか、各 PVCS ファイルとディレクトリのグループ所有権を PVCS グループに変更します。

Serena 管理者は、コンフィグレーションファイル、アクセスコントロールデータベース、nfsmap ファイル、および Version Manager 実行ファイルを含むすべてのディレクトリ、およびそれらのディレクトリ中のファイルのすべてを所有している必要があります。

ユーザ ID とグループの作成については、オペレーティングシステムのマニュアルを参照してください。

- 2 クロスプラットフォーム環境で Version Manager を使用する場合は、setuid モードを終了します。
- 3 Windows と UNIX システムの両方のユーザがコンフィグレーションファイルを共有できるよう、UNIX 上で nfsmap ファイルを編集して Windows のドライブ番号をそれに対応する UNIX のパス名にマップします。43 ページの『nfsmap ファイルの編集』を参照してください。



注

- この手順はアーカイブを作成する前に行う必要があります。
- nfsmap ファイルのエントリ数は 26 有効 (コメントではない) エントリに制限されています。それ以上のエントリは無視されます。

- 4 UNIX 上で、管理者および各ユーザは、nfsmap ファイルおよび Windows では VMWFVC.DLL、UNIX では vmufvc.a を含むディレクトリの場所に、PVCS_BINDIR 環境変数を設定する必要があります。

5 UNIX と Windows で適切な権限を割り当てます。

■ UNIX の場合 :

ユーザ	割り当てる権限	対象
PVCS ユーザ および管理者	読み込み、書き込み、実行	コンフィグレーションファイルで指定されたディレクトリ
PVCS ユーザ	読み込み、実行	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ
PVCS ユーザ	読み込み	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル
<p>注 ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。</p>		
PVCS 管理者	読み込み、書き込み、実行	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ
PVCS 管理者	読み込み、書き込み	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル

■ Windows の場合 :

ユーザ	割り当てる権限	対象
PVCS ユーザ および管理者	すべての権限	コンフィグレーションファイルで指定されたディレクトリ
PVCS ユーザ	読み込み	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ
<p>注 ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。</p>		
PVCS 管理者	すべての権限	マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリ

6 Version Manager を次のように設定します。

- 新規アーカイブに対しては NoWriteProtect ディレクティブ、既存のアーカイブに対しては `vcs -pw` コマンドを使用して、アーカイブを書き込み可能とする。
- 新規アーカイブに対しては Translate ディレクティブ、既存のアーカイブに対しては `vcs +pt` コマンドを使用して EOL シーケンスを変換する。
- データの破損を防ぐため、バイナリファイルの新規アーカイブに対しては NoTranslate ディレクティブ、既存のアーカイブに対しては `vcs -pt` コマンドを使用して、バイナリファイルの EOL シーケンスが変換されないようにする。
- Windows と UNIX でユーザ ID が同様に読み込まれるよう、NoCase=VCSID ディレクティブを使用して、ユーザ ID の大文字と小文字が区別されないように設定する。

- 7 Windows と UNIX の間で、共有されるファイルとディレクトリの名称で、大文字と小文字の区別を一致させます。

setuid の無効化

UNIX 上では、次のコマンドを使用して Version Manager 実行ファイルの setuid 権限を禁止します。

```
chmod u-s executable
```

ここで *executable* は次のいずれかです。

■ get	■ regen	■ vjournal
■ ident	■ vcompres	■ vlog
■ makedb	■ vconfig	■ vmrg
■ put	■ vcs	■ vpromote
■ pvcsvmsuid	■ vdel	■ vsql
■ readdb	■ vdiff	

nfsmap ファイルの編集

Version Manager には、Windows と UNIX システムでユーザがコンフィグレーションファイルを共有できるように、nfsmap ファイルが用意されています。nfsmap ファイルは、Windows のドライブ番号または UNC パス名を、それに対応する UNIX パス名にマップします。このファイルは UNIX 上で保存されます。このファイルが存在する場合は、Version Manager はコンフィグレーションファイル内のパス名の読み込みおよび書き込み時に MS-DOS パスフォーマットを使用します。



注 この手順は、アーカイブを作成する前に完了する必要があります。

UNIX 上で nfsmap ファイルを編集するには

- 1 nfsmap ファイルをテキストエディタで開きます。nfsmap ファイルのデフォルトの場所は、以下のとおりです。

```
Install_Location/vm/common/bin/OS/nfsmap
```

ここで、*os* はオペレーティングシステムの名前です。たとえば、Solaris の場合、パスは次のようになります。

```
Install_Location/vm/common/bin/solaris/nfsmap
```

- 2 各マッピングをそれぞれの行の 2 列を使用して入力します。左の列にはドライブ文字または UNC パス名を、右の列にはそれに対応する UNIX ディレクトリを入力します。UNC パス名にスペースが含まれる場合は、それを二重引用符 (") で囲む必要があります。

例 1 : M: ドライブを UNIX ディレクトリである /product/dev にマップするには、次のように入力します。

```
M /product/dev
```

例 2 : UNC パス名 "¥¥myserver¥vol2 abc" を UNIX ディレクトリである /product/dev にマップするには、次のように入力します。

```
"¥¥myserver¥vol2 abc" /product/dev
```



重要! ドライブ文字、UNC パス、UNIX ディレクトリはそれぞれ一意である必要があります。同じドライブ文字、UNC パス、UNIX ディレクトリを別のマッピングで繰り返し使用することはできません。

3 変更を保存し、テキストエディタを終了します。

PVCS_BINDIR の設定

UNIX 上で、PVCS_BINDIR 環境変数を nfsmap ファイルとアクセスコントロールデータベースのある場所に設定します。



注 この変数は /etc/profile と /etc/.login に設定するか、あるいは各ユーザの .profile および .cshrc ファイルに設定するのが最適です。

PVCS_BINDIR 環境変数を設定するには

ユーザのシェルに応じて、次のいずれかのコマンドを入力します。

csh の場合: `setenv PVCS_BINDIR directory`

または、

sh、ksh、bash の場合: `PVCS_BINDIR=directory; export PVCS_BINDIR`

ここで *directory* は、nfsmap ファイルとアクセスコントロールデータベースのある場所です。通常は、これらのファイルは UNIX の Version Manager インストールディレクトリにあります。デフォルトのディレクトリは /usr/pvcs です。



注 vcompress、vdel、vdiff、vsq1、readdb、および makedb の 6 つのコマンドラインコマンドは nfsmap ファイルを読み込みません。

権限の割り当て

UNIX と Windows の両方において、ユーザに次のように権限を割り当てます。

- ユーザは、アーカイブの読み込みと書き込みが可能である、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル (nfsmap ファイルは UNIX 上のみ) については読み込みのみが可能。ユーザには Version Manager の設定を変更する可能性があるため、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルに対する書き込み権は割り当てない。



注 ユーザが自分のパスワードを変更できるようにするには、PVCS ユーザにアクセスコントロールデータベースへの読み込みおよび書き込み権限が必要です。

- 管理者は、アーカイブ、マスターコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイル (nfsmap ファイルは UNIX 上のみ) の読み込みが可能。

UNIX および Windows で権限を割り当てるには

- 1 UNIX 上で、PVCS ユーザと Serena 管理者に、コンフィグレーションファイル内で指定されたディレクトリ（および該当するサブディレクトリ）に対する読み込み、書き込み、および実行の権限を割り当てます。このディレクトリには、アーカイブディレクトリ（VCSDir）、セマフォディレクトリ（SemaphoreDir）、内部使用の一時ファイルが含まれるディレクトリ（WorkDir）、一時使用のアーカイブファイルディレクトリ（ArchiveWork）、参照ディレクトリ（ReferenceDir および VCSDir）、およびジャーナルファイルディレクトリ（Journal）が含まれます。

それぞれのディレクトリについて、次のコマンドを入力します。

```
chmod ug+rwx directory
```

- 2 UNIX 上で、PVCS ユーザに Version Manager のコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルを含むディレクトリに対する読み込みと実行の権限を割り当てます。また、Serena 管理者には、このディレクトリに対する読み込み、書き込み、および実行の権限を割り当てます。

それぞれのディレクトリについて、次のコマンドを入力します。

```
chmod ug+rx,u+w directory
```

- 3 UNIX 上で、PVCS ユーザにコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルについて読み込みの権限のみを割り当てます。また、Serena 管理者には、これらのファイルについて読み込みと書き込みの権限を割り当てます。

それぞれのディレクトリに移動し、それぞれのファイルについて次のコマンドを入力します。

```
chmod ug+r,u+w file_name
```

- 4 Windows 上で、PVCS ユーザと Serena 管理者に、コンフィグレーションファイル内で指定されたディレクトリ（および該当するサブディレクトリ）に対するすべての権限を割り当てます。このディレクトリにはアーカイブディレクトリ（VCSDir）、セマフォディレクトリ（SemaphoreDir）、ワークディレクトリ（WorkDir）、アーカイブワークディレクトリ（ArchiveWork）、および参照ディレクトリ（ReferenceDir および VCSDir）が含まれます。

- 5 Windows 上で、PVCS ユーザに、Version Manager のコンフィグレーションファイル、アクセスコントロールデータベース、および nfsmap ファイルと、これらを含むディレクトリに対する読み込み権限のみを割り当てます。ユーザには、これらのファイルの変更は許可しません。

- 6 Windows 上で、手順 5 に記載されたファイルに対するすべての権限を Serena 管理者に割り当てます。管理者にはこれらのファイルの変更を許可します。

Version Manager の設定

このセクションでは、Version Manager を次のように設定します。

- アーカイブを書き込み可能にする。
- テキストファイルの EOL シーケンスを変換し、バイナリファイルの EOL シーケンスは変換しない。
- ユーザ ID の大文字と小文字を区別しないようにする。

アーカイブへの書き込み許可

Windows と UNIX でアーカイブを共有する場合は、アーカイブファイルが書き込み可能となっている必要があります。

- 1 次の行をマスターコンフィグレーションファイルに記入し、新規作成のアーカイブが書き込み可能となるよう Version Manager を設定します。

```
NoWriteProtect
```

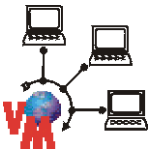
- 2 次の行をマスターコンフィグレーションファイルに記入し、マスターコンフィグレーションファイルの WriteProtect および NoWriteProtect ディレクティブを禁止します。

```
Disallow WriteProtect, NoWriteProtect
```

- 3 各アーカイブに対して次のコマンドを使用し、既存のすべての Version Manager アーカイブを書き込み可能とします。

```
vcs -pw archive_name
```

EOL シーケンスの変換



UNIX と Windows でアーカイブを共有するには、テキストファイルのリビジョンをチェックインするときに、UNIX 上の EOL シーケンスをキャリッジリターン (CR) からキャリッジリターン + ラインフィード (CR+LF) に変換し、リビジョンをチェックアウトするときには逆に変換するよう、各プロジェクトデータベースを設定する必要があります。



注意！ バイナリファイルの EOL シーケンスを変換すると、作業ファイルが破損することがあります。

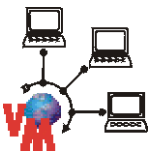
次の行をマスターコンフィグレーションファイルに記入します。

```
NoTranslate  
Translate .ext
```

ここで `.ext` はアーカイブに入れるテキストファイルのファイル拡張子です。アーカイブに入れるテキストファイルの各タイプごとに `Translate` の行を入力する必要があります。このディレクティブで指定できるのは、1 行ごとに 1 つのファイル拡張子のみです。`Translate` ディレクティブの詳細については『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

`Translate` ディレクティブはアーカイブを新規作成するときのデフォルトのみを定義し、既存のアーカイブには影響しません。既存のアーカイブに対しては、テキストファイルについて変換を行うには `vcs +pt` コマンド、バイナリファイルが変換されないようにするには `vcs -pt` コマンドを使用します。

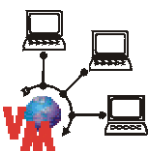
ユーザ ID での大文字と小文字の区別



UNIX と Windows でユーザ ID が同じように読み込まれるようにするには、次の行をマスターコンフィグレーションファイルに記入します。

```
NoCase=VCSID
```

ファイル名およびディレクトリ名の大文字と小文字の使用



Windows と UNIX で共有されるファイルとディレクトリの名称において、大文字と小文字が一貫して使用されるよう、次のいずれかを行います。

- デフォルトの作業ファイル名で使用される大文字と小文字が正しくなるよう、新規アーカイブはすべて UNIX で作成する。

-
- Windows で作成されたアーカイブに、VCS コマンドを使用して、アーカイブ中の作業ファイル名で使用される大文字と小文字を修正する。小文字の使用をお勧めします。

次のステップ

Version Manager のセットアップが完了したら、次の作業を行うことができます。

- Version Manager の設定。69 ページの『[Version Manager の設定](#)』を参照してください。
- セキュリティの追加。201 ページの『[セキュリティの使用](#)』を参照してください。
- プロモーションモデルの定義。245 ページの『[プロモーションモデルの使用](#)』を参照してください。
- イベントトリガの定義。271 ページの『[イベントトリガの使用](#)』を参照してください。

第 3 章

Version Manager の設定

はじめに	69
コンフィグレーションオプションについて	70
デスクトップクライアントのコンフィグレーションファイル	75
コマンドラインインターフェイスでのコンフィグレーションファイル	79
コンフィグレーションオプションの設定	81
Version Manager へのマスターコンフィグレーションファイルの埋め込み	118
TrackerLink と SourceBridge のセットアップ	120
Serena ChangeMan Mover の設定	125
トラブルシューティング	127

はじめに

15 ページの『デスクトップクライアントを使用した Version Manager の設定』、または 53 ページの『コマンドラインインターフェイスを使用した Version Manager のセットアップ』で説明したように、基本的なプロジェクトモデルを設定した後、Version Manager のデフォルトの動作が目的にそぐわない場合はその動作を定義できます。デスクトップクライアントでの Version Manager のデフォルトの動作は 72 ページの『デスクトップクライアントでのデフォルト設定』、コマンドラインインターフェイスでのデフォルト動作は 74 ページの『コンフィグレーションファイルが使用されない場合のデフォルト設定』に説明されています。

Version Manager は必要に応じてプロジェクトデータベースレベルまたはプロジェクトレベル、あるいはその両方で設定することができます。たとえば、次のように設定できます。

- すべてのプロジェクトデータベースの動作を同じにするには、Version Manager にコンフィグレーションファイルを埋め込む。
- 特定のプロジェクトデータベース内のすべてのプロジェクトの動作を同じにするには、プロジェクトレベルの代わりにプロジェクトデータベースに関連するコンフィグレーションファイルを設定する。
- プロジェクトデータベース内で一部のプロジェクトの動作を変えるには、それらのプロジェクトに対して固有のコンフィグレーションファイルを作成する。たとえば、一部のプロジェクトにはプロモーションモデルを定義し、他には定義しないことができます。

この章ではまず、コンフィグレーション設定のオプションについて説明し、用意されているオプションの表を示します。次に、当初のデフォルト設定が使用可能かどうかを判断するため、オプションのデフォルトを示します。3 番目にコンフィグレーションファイル、および Version Manager がそれをどのように使用するかについて説明します。最後に、すべてのプロジェクトコンフィグレーションオプションについて使用可能な設定を示し、デスクトップクライアントおよびコマンドラインインターフェイスを使用してどのようにそれらを設定するかを説明します。

コンフィグレーションオプションについて

コンフィグレーションオプションとは、Version Manager の動作を規定する一連の設定です。たとえば、作業ファイルがチェックイン後に削除されるかどうか、どのアクセスコントロールデータベースが使用されるかなどは、コンフィグレーションオプションによって制御されます。

コンフィグレーションオプションは、コンフィグレーションファイル内で設定され、単独のコンフィグレーションファイルあるいは複数のコンフィグレーションファイルについて設定を行うことができます。

コンフィグレーションファイルの詳細については、75 ページの『デスクトップクライアントのコンフィグレーションファイル』および 79 ページの『コマンドラインインターフェイスでのコンフィグレーションファイル』を参照してください。

表 3-1 に使用可能なコンフィグレーションオプションを示します。

表 3-1 : コンフィグレーションオプション

オプション	目的
アーカイブの作成	<ul style="list-style-type: none">■ アーカイブを排他的にロックする。■ アーカイブの所有権を割り当てる。■ アーカイブにアクセスできるユーザを指定する。■ アーカイブを読み取り専用ファイルにする。■ ファイルにチェックインするときに Version Manager がファイルのアーカイブを見つけれない場合は、自動的にアーカイブを作成する。
ブランチ	<ul style="list-style-type: none">■ ブランチが開始されるリビジョンを識別するためのバージョンラベルを指定する。■ ブランチリビジョンに割り当てるバージョンラベルを指定する。■ リビジョンが指定されていない場合に使用するリビジョンを特定する。■ ブランチが作成される前にユーザに警告を発する。
ロック	<ul style="list-style-type: none">■ ユーザに複数のリビジョンをロックすることを許可する。■ リビジョンごとに複数のロックを許可する。■ 変更されていないリビジョンを保存する場合にロックを解除する。
プロモーションモデル	<ul style="list-style-type: none">■ プロモーションモデルを定義する。
アクセスコントロールデータベース	<ul style="list-style-type: none">■ アクセスコントロールデータベースを指定する。■ アクセスコントロールデータベースを使用できるようにする。
ジャーナルファイル	<ul style="list-style-type: none">■ アーカイブの操作に関するログを保存する。
ログインソース	<ul style="list-style-type: none">■ ユーザ ID ソースを指定する。■ アクセスコントロールデータベースにユーザが自動的に作成されるかどうかを指定し、作成される場合には各ユーザに割り当てられるデフォルトの権限を指定する（デスクトップクライアントのみ）。
作業ファイルの属性	<ul style="list-style-type: none">■ 作業ファイルを保存後に削除する。■ 作業ファイルを保存後に、その書き込み禁止コピーを保存する。■ 作業ファイルを保存後に、その書き込み許可コピーを保存する。

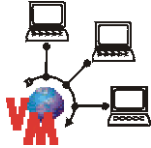
オプション	目的
キーワード展開	<ul style="list-style-type: none"> ■ キーワードの展開に使用されるパス区切り文字を指定する。 ■ キーワードの展開時にタイムスタンプを更新する。
参照用ディレクトリ	<ul style="list-style-type: none"> ■ 現在の作業ファイルのコピーが自動的に保存されるディレクトリを指定する。 ■ 各アーカイブディレクトリごとに参照用ディレクトリを保持する。 ■ トランクリビジョンのみを保存する。 ■ 参照用ディレクトリ中のファイルを書き込み禁止にする。
アーカイブ検索パス (コマンドラインインターフェイスのみ)	<ul style="list-style-type: none"> ■ ファイルに対して作業を行うときに、Version Manager が検索する場所を指定する。 ■ ワイルドカードを指定した場合に、最初に一致するものが見つかったアーカイブディレクトリ以外のディレクトリにあるものも一致するものと見なすかどうかを指定する。 ■ 作業ファイル名のみが指定されている場合に、Version Manager のアーカイブの検索方法を指定する。
コマンドライン (コマンドラインインターフェイスのみ)	<ul style="list-style-type: none"> ■ 著作権とバージョンを表示する。 ■ 詳細なメッセージを表示する。 ■ コマンドラインのエディタを指定する。 ■ テンポラリファイルのディレクトリを指定する。 ■ メッセージファイルの拡張子を指定する。 ■ メッセージファイルを読んだ後に削除する。
セマフォ	<ul style="list-style-type: none"> ■ セマフォファイルが作成されるディレクトリを指定する。 ■ セマフォファイルに使用される拡張子を指定する。 ■ アーカイブにアクセスを試みる回数を指定する。 ■ 共有アーカイブにアクセスを試みる間隔を指定する。
テンポラリファイル	<ul style="list-style-type: none"> ■ テンポラリアーカイブファイルのディレクトリを指定する。 ■ テンポラリ内部ファイルのディレクトリを指定する。
ファイルタイプ	<ul style="list-style-type: none"> ■ アーカイブの拡張子を指定する。 ■ デルタを保存する。 ■ EOL シーケンスを変換する。 ■ キーワードを展開する。 ■ キーワード展開時のコメント接頭辞を指定する。 ■ キーワード展開時の改行コードを指定する。 ■ ファイル比較時の桁を指定する。 ■ マスクされた桁に番号を付け直す。 ■ 論理レコードの長さを指定する。
イベントトリガ	<ul style="list-style-type: none"> ■ Version Manager イベントの前または後に、ユーザ定義の命令を実行する。
許可 / 禁止	<ul style="list-style-type: none"> ■ ユーザがコンフィグレーションオプションの設定を変更できないようにする。

デスクトップクライアントでのデフォルト設定

プロジェクトが作成されるときに、デフォルトではマスターコンフィグレーションファイルが作成され、プロジェクトデータベースに関連付けられます。

このマスターコンフィグレーションファイルは、Version Manager のインストール時にインストールされる default.cfg ファイルをコピーして作られます。デフォルトではこのファイルは次の場所に置かれています。

- Windows の場合：
`drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合：`/usr/serena/vm/common/pvcsprop/pvcs/vm`



注 ファイルサーバに作成したプロジェクトデータベースの場合、デフォルトのコンフィグレーションファイルは、defaultfs.cfg という名前になります。

default.cfg ファイル中の設定は必要に応じて変更でき、Version Manager はこの変更されたファイルを使用して新しいコンフィグレーションファイルを作成します。

このファイル中のコンフィグレーションオプションの設定を変更せず、プロジェクトコンフィグレーションファイルを定義しない場合は、Version Manager はこのコンフィグレーションファイル中の設定に従って動作します。このファイルの設定は、プロジェクトデータベースを選択し、[管理] - [詳細設定] を選択するか、テキストエディタでファイルを開いて変更できます。[プロジェクトの詳細設定] ダイアログボックス中の設定は、コンフィグレーションファイル中の設定を反映しています。Version Manager は次のように動作します。

- アーカイブ拡張子は +-arc (例：text.txt-arc)。112 ページの『[アーカイブ拡張子の変更](#)』を参照してください。
- アーカイブが存在しない場合は、作業ファイルのチェックイン時に自動的に作成される。
- アーカイブは書き込み禁止。
- ブランチが作成されるときは、ユーザに警告が表示される。
- 作業ファイルはチェックイン時に削除されない。
- バイナリファイルではキーワードは展開されない。テキストファイルでは、特定の拡張子を持ったファイルの場合、キーワードが展開されます。この動作は新規に作成されるアーカイブにのみ適用され、既存のアーカイブには影響しません。
- バイナリファイルでは EOL シーケンスは変換されない。UNIX テキストファイルでは、特定の拡張子を持ったファイルの場合、変換されます。この動作は新規に作成されるアーカイブにのみ適用され、既存のアーカイブには影響しません。
- 単一のリビジョンに対して複数のロックを設定することは認められない。
- 1 人のユーザが、アーカイブ中の複数のリビジョンに対してロックを設定できる。
- 1 つのアーカイブ中では複数のリビジョンに対してロックを設定できる。
- ファイルセマフォが指定される。
- セマフォの拡張子は +-sem (例：text.txt-sem)。
- アーカイブのセマフォは、アーカイブと同じディレクトリに作成される。
- Version Manager によるアーカイブへのアクセスのリトライ回数は 6 回。
- Version Manager によるアーカイブへのアクセスのリトライ間隔は 3 秒。
- **ファイルサーバ以外のプロジェクトデータベースの場合**：ログインソースはホスト。デスクトップクライアントでは、ユーザが存在しない場合は、アクセスコントロールデータベース中に自動

的に作成され、Unlimited 権限セットが割り当てられます。

ファイルサーバプロジェクトデータベース：ログインソースは VLOGIN です。ユーザはアクセスコントロールデータベースに自動的に追加されません。

- **ファイルサーバ以外のプロジェクトデータベースの場合**：ユーザ ID では、大文字と小文字は区別されない。
ファイルサーバプロジェクトデータベースの場合：ユーザ ID では、大文字と小文字は区別される。
- チェックインがキャンセルされた場合は、変更されていないリビジョンに対するロックは解除される。
- リビジョンは、そのファイルをロックしたユーザによってチェックインされなければならない。
- バイナリファイルの場合、リビジョンはデルタセットとしては保存されない。リビジョンは、特定の拡張子を持つテキストファイルの場合に、デルタセットとして保存されます。この動作は新規に作成されるアーカイブにのみ適用され、既存のアーカイブには影響しません。
- Version Manager がチェックイン時にキーワードを展開するときは、タイムスタンプは更新されない。
- チェックイン後の作業ファイルのコピーは、参照用ディレクトリには置かれない。
- CTRL+Z の後の文字は無視されない。
- 自動ブランチは設定されない。
- **ファイルサーバ以外のプロジェクトデータベースの場合**：有効なアクセスコントロールデータベースはない。
ファイルサーバプロジェクトデータベースの場合：アクセスコントロールは有効。
- ジャーナルファイルは保存されない。
- キーワード展開時には、スラッシュ (/) がディレクトリの区切り文字として使用される。
- 作業ファイルをチェックインするには、リビジョンにロックを設定する必要がある。

以下の動作は、コマンドラインインターフェイスにのみ適用されます。

- メッセージファイルは読んだ後も削除されない。
- メッセージファイルの拡張子は +-msg (例：text.txt-msg)。
- 著作権とバージョンは表示される。
- 詳細なメッセージが表示される。
- VCSDir はプロジェクトのアーカイブのルートに設定され、その下のディレクトリはすべてアーカイブのディレクトリとして取り扱われる。
- アーカイブを探すときには、指定された作業ファイルパスが使用される。
- デフォルトのエディタは指定されていない。
- デフォルトの日付フォーマットは mm/dd/yyyy hh:mm:ss (年/日/月 時:分:秒)。
- 月と午前/午後の表示には、January (1 月)、February (2 月)、March (3 月)、April (4 月)、May (5 月)、June (6 月)、July (7 月)、August (8 月)、September (9 月)、October (10 月)、November (11 月)、December (12 月) および AM (午前)、PM (午後) が使用される。

コンフィグレーションファイルが使用されない場合のデフォルト設定

コマンドラインインターフェイスでは、Version Manager にデフォルトのコンフィグレーションファイルは設定されていません。ただし、製品にはデフォルトのコンフィグレーションファイル、default.cfg が添付されています。デフォルトでは、このファイルは次の場所に置かれています。

- Windows の場合 : `drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合 : `/usr/serena/vm/common/pvcsprop/pvcs/vm`

コマンドラインインターフェイスで、このファイルをマスターコンフィグレーションファイルとして使用するには、このファイルのコピーを作成し、必要に応じてオプションを変更し、さらにコピーされたファイルを VCONFIG コマンドにより Version Manager に埋め込んでください。

コマンドラインインターフェイスを使用するときに Version Manager にコンフィグレーションファイルを関連付けない場合、またはデスクトップクライアントを使用するときにプロジェクトデータベースまたはプロジェクトからコンフィグレーションファイルの関連付けを解除した場合は、Version Manager は次のように動作します。

- アーカイブ拡張子のテンプレートは `??V__` となり、アーカイブのファイル名は `text.txv` のようになる。[112 ページの『アーカイブ拡張子の変更』](#)を参照してください。
- アーカイブが存在しない場合は、作業ファイルのチェックイン時に自動的に作成される。
- アーカイブは書き込み禁止。
- Version Manager は、現在のアーカイブディレクトリ内でアーカイブを探す。
- 参照用ディレクトリは指定されない。
- ブランチが作成されるときに、ユーザには警告は表示されない。
- 作業ファイルはチェックイン時に削除される。
- キーワードは展開されない。
- EOL シーケンスは、`.c`、`.h`、`.pas`、`.mak`、`.for`、`.bas`、`.asm`、`.bat` を除き、すべてのファイルタイプについて変換されない。
- 単一のリビジョンに対して複数のロックを設定することは認められない。
- 1 人のユーザは、アーカイブ中の複数のリビジョンに対してロックを設定できない。
- 1 つのアーカイブ中では複数のリビジョンに対してロックを設定できる。
- セマフォの拡張子は `??$__` (例 : `text.tx$`)。
- Version Manager によるアーカイブへのアクセスのリトライ回数は 3 回。
- Version Manager によるアーカイブへのアクセスのリトライ間隔は 1 秒。
- ログインソースは Host。
- 変更されていないリビジョンに対するロックは解除されない。
- アーカイブは圧縮されない。
- デルタは圧縮されない。
- アーカイブに配置される当初のファイルのコピー (作業イメージ) は圧縮されない。
- メッセージファイルは読んだ後も削除されない。
- 作業ファイルをチェックインするには、リビジョンにロックを設定する必要がある。
- リビジョンは、そのファイルをロックしたユーザによってチェックインされなければならない。

- リビジョンはデルタのセットとして保存される。
- アーカイブを探すときは作業ファイルパスが使用される。
- Version Manager がチェックイン時にキーワードを展開するときに、タイムスタンプが更新される。
- CTRL+Z の後の文字は無視されない。

デスクトップクライアントのコンフィグレーションファイル

設定に関するオプションは、コンフィグレーションファイルに保存されます。デスクトップクライアントには、次の 2 つのタイプのコンフィグレーションファイルがあります。

- マスターコンフィグレーションファイルには、プロジェクトデータベースとそのすべてのプロジェクトに対するコンフィグレーションオプションが含まれます。
- プロジェクトコンフィグレーションファイルには、プロジェクトに対するコンフィグレーションオプションが含まれており、マスターコンフィグレーションファイルで禁止されていないかぎり、マスターコンフィグレーションファイル中の設定より優先されます。個々のプロジェクトに対してコンフィグレーションファイルを設定する必要はなく、マスターコンフィグレーションファイル中の設定を使用できます。

Version Manager デスクトップクライアントは、次の場合にコンフィグレーションファイルを使用します。

- プロジェクトデータベースまたはプロジェクトに関連付けられている場合。プロジェクトデータベースに関連付けられたコンフィグレーションファイルは、マスターコンフィグレーションファイルと呼ばれます。
- Version Manager に埋め込まれている場合。[118 ページの『Version Manager へのマスターコンフィグレーションファイルの埋め込み』](#)を参照してください。

マスターコンフィグレーションファイル

マスターコンフィグレーションファイルには、ほかのコンフィグレーションファイルによってリセットされないオプションを設定でき、Version Manager の動作を標準化することができます。マスターコンフィグレーションファイル中でオプションを変更禁止に設定すると、Version Manager はマスターコンフィグレーションファイルの設定に従って動作します。

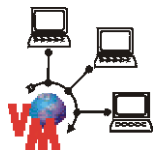
たとえば、マスターコンフィグレーションファイルで新規ファイルのチェックイン時に自動的にアーカイブを作成するオプションを設定し、次にこのオプションを変更禁止に設定すると、このオプションは、ほかのコンフィグレーションファイルからは変更できず、Version Manager は自動的にアーカイブを作成します。つまり、マスターコンフィグレーションファイル中のオプションを設定し、さらにそのオプションを変更禁止に設定するとほかのコンフィグレーションファイルではそのオプションを変更することができません。

デフォルトのマスターコンフィグレーションファイル

プロジェクトデータベースを作成すると、自動的にマスターコンフィグレーションファイルが作成されます。このファイルの設定は必要に応じて変更できます。セキュリティ上の理由のため、このコンフィグレーションファイルの名称は暗号化されます。たとえば、このファイルには c6bonpj1.cfg などの名前が付けられます。デフォルトでは、このファイルはアーカイブディレクトリに置かれます。

このマスターコンフィグレーションファイルは、Version Manager のインストール時にインストールされる default.cfg ファイルをコピーして作成されます。デフォルトでは、このファイルは Version Manager のインストール時に次の場所に置かれます。

- Windows の場合 : `drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合 : `/usr/serena/vm/common/pvcsprop/pvcs/vm`



注 ファイルサーバに作成したプロジェクトデータベースの場合、デフォルトのコンフィグレーションファイルは、defaultfs.cfg という名前になります。

default.cfg ファイルの設定は必要に応じて変更でき、新規マスターコンフィグレーションファイルの作成は、変更されたファイルに基づいて行われます。

マスターコンフィグレーションファイルの埋め込み

オプションとして、Version Manager にマスターコンフィグレーションファイルを埋め込むこともできます。このファイルは、すべてのプロジェクトデータベースに対する Version Manager の動作をコントロールします。Version Manager にコンフィグレーションファイルを埋め込むと、そのファイルがマスターコンフィグレーションファイルになり、各プロジェクトデータベースに関連付けられたコンフィグレーションファイルは、マスターコンフィグレーションファイルでなく、プロジェクトコンフィグレーションファイルとなります。したがって、プロジェクトコンフィグレーションファイル中の変更禁止オプションはすべて無視されます。



注 default.cfg ファイルは、プロジェクトデータベースに対するコンフィグレーションファイルの作成に使用されるため、これを Version Manager に埋め込むことはしないでください。埋め込むと、埋め込まれたマスターコンフィグレーションファイルと、プロジェクトデータベースのコンフィグレーションファイルの設定が、最初は同じになります。

Version Manager にコンフィグレーションファイルを埋め込むことにより、すべてのユーザが同一のコンフィグレーション設定を使用し、また、ユーザが別のマスターコンフィグレーションファイルを使用できないことが保証されます。

Version Manager に埋め込まれたマスターコンフィグレーションファイルは、それが埋め込まれた Version Manager を使用するすべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。マスターコンフィグレーションファイルを埋め込む手順については、[118 ページの『Version Manager へのマスターコンフィグレーションファイルの埋め込み』](#)を参照してください。

デスクトップクライアントでのコンフィグレーションオプション設定のガイドライン

このセクションでは、デスクトップクライアントでのマスターおよびプロジェクトコンフィグレーションファイルに設定する、オプションのタイプに関するガイドラインを示します。

マスターコンフィグレーションファイル

マスターコンフィグレーションファイルは、Version Manager のプロジェクトデータベースに対する動作を規定し、ほかのコンフィグレーションファイルによっては変更できないオプションを指定（禁止に設定）します（前述のマスターコンフィグレーションファイルに関する説明を参照してください）。次にその例を示します。

- Version Manager がユーザを識別するためのログインソース。
- アクセスコントロールデータベースなど、すべてのユーザがアクセスできないプログラムファイルの場所。

- リビジョンへのアクセスと並行開発に影響するロックのオプション。
- 同時にアクセスが発生した場合の、Version Manager によるアーカイブの破損防止に影響するセマフォオプション。
- すべてのプロジェクトが使用する必要のあるプロモーションモデル。

プロジェクトコンフィグレーションファイル

プロジェクトコンフィグレーションファイルは、マスターコンフィグレーションファイルで変更禁止に設定されていない、プロジェクトごとのオプションの設定に使用します。プロジェクトコンフィグレーション設定は、マスターコンフィグレーションファイルの設定に優先します。各プロジェクトに対してオプションを設定し、Version Manager がどのようにそのプロジェクトに対して動作するかを決定します。次にその例を示します。

- プロジェクト用にカスタマイズされたプロモーションモデル。
- プロジェクトごとのジャーナルファイル。
- プロジェクトごとの参照用ディレクトリ。

デスクトップクライアントによるコンフィグレーションファイルの読み込み

Version Manager は常に、プロジェクトコンフィグレーションファイルを読み込む前に、マスターコンフィグレーションファイルを読み込みます。これは、マスターコンフィグレーションファイルには、ほかのコンフィグレーションファイルでは再設定できないオプションが定義されているためです。

プロジェクトコンフィグレーションファイルの設定は、そのオプションがマスターコンフィグレーションファイルによって変更禁止に設定されていないかぎり、マスターコンフィグレーションファイルの設定より優先されます。たとえば、マスターコンフィグレーションファイルでアーカイブ拡張子が +rev に設定され、さらにこのオプションが変更禁止に設定されていない場合に、プロジェクトコンフィグレーションファイルでアーカイブ拡張子が _src に設定されると、そのプロジェクトコンフィグレーションファイルの設定が使用されます。



注 プロジェクトコンフィグレーションファイルの設定がマスターコンフィグレーションファイルの設定に優先することの例外として、プロモーションオプションがあります。[248 ページの『プロモーションモデルの定義』](#)を参照してください。

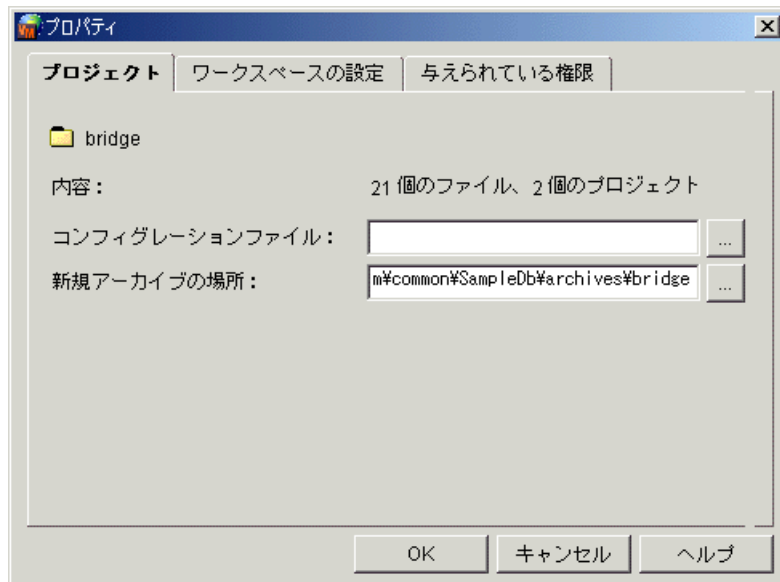
Version Manager がマスターコンフィグレーションファイルまたはプロジェクトコンフィグレーションファイルを見つけられない場合は、各オプションについてデフォルトの設定が使用されます。[74 ページの『コンフィグレーションファイルが使用されない場合のデフォルト設定』](#)を参照してください。

コンフィグレーションファイルのプロジェクトデータベースおよびプロジェクトへの関連付け

デスクトップクライアントのデフォルトでは、作成されるプロジェクトデータベースごとにマスターコンフィグレーションファイルが関連付けられます。プロジェクトデータベースを作成するときには、必要に応じて既存のコンフィグレーションファイルを使用できます。

プロジェクトは、デフォルトではプロジェクトデータベースに関連付けられていません。プロジェクトにプロジェクトデータベースを関連付けるには、次の手順を実行します。

- 1 コンフィグレーションファイルに関連付けるプロジェクトを選択します。
- 2 [ファイル] - [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示され、[プロジェクト] タブがアクティブになっています。



- 3 〈コンフィグレーションファイル〉フィールドに、プロジェクトに関連付けるコンフィグレーションファイルの場所と名称を入力します。コンフィグレーションファイルが存在しない場合は、ここに指定した場所と名称により自動的に作成されます。デフォルトの名称は project.cfg です。このコンフィグレーションファイルではオプションが設定されていないので、この後でプロジェクトを設定する必要があります。81 ページの『[コンフィグレーションオプションの設定](#)』を参照してください。
- 4 【OK】をクリックします。

設定されたプロジェクトデータベースとプロジェクトでは、次の作業が必要となる場合があります。

- コンフィグレーションファイルの名称変更。
- コンフィグレーションファイルの移動。
- 複数のプロジェクトデータベースの 1 つのマスターコンフィグレーションファイルへの関連付け、および複数のプロジェクトの 1 つのプロジェクトコンフィグレーションファイルへの関連付け。
- すべてのコンフィグレーションファイルの、1 か所の安全な場所への保存。

コンフィグレーションファイル名の変更

プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイルの名称を変更するには、次の手順を実行します。

- 1 コンフィグレーションファイルのコピーを作成します。
- 2 コンフィグレーションファイルの名称を変更します。
- 3 Version Manager デスクトップクライアントから、名称を変更されたコンフィグレーションファイルを、該当するプロジェクトデータベースまたはプロジェクトに関連付けます（[ファイル] - [プロパティ]）。

- 4 必要に応じて、古いコンフィグレーションファイルを削除します。

コンフィグレーションファイルの移動

プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイルを移動するには、次の手順を実行します。

- 1 コンフィグレーションファイルのコピーを作成します。
- 2 コピーされたファイルを新しい場所に移動します。
- 3 Version Manager デスクトップクライアントから、コピーされたコンフィグレーションファイルを、該当するプロジェクトデータベースまたはプロジェクトに関連付けます（[ファイル] - [プロパティ]）。
- 4 必要に応じて、古いコンフィグレーションファイルを削除します。

コマンドラインインターフェイスでのコンフィグレーションファイル

コンフィグレーションオプションは、コンフィグレーションファイルに保存されます。コマンドラインインターフェイスには、次の 2 つのタイプのコンフィグレーションファイルがあります。

- マスターコンフィグレーションファイル。Version Manager がすべてのアーカイブとコマンドに対し、どのように動作するかを規定するコンフィグレーションオプションが含まれます。
- ローカルコンフィグレーションファイル。コンフィグレーションファイルのあるディレクトリに保存されたファイルに対するコンフィグレーションオプションが含まれます。このファイルは、デフォルトでは `vcs.cfg` という名称です。

Version Manager のコマンドラインインターフェイスは、次の場合にコンフィグレーションファイルを使用します。

- Version Manager に埋め込まれている場合。埋め込まれたコンフィグレーションファイルはマスターとなります。[118 ページの『Version Manager へのマスターコンフィグレーションファイルの埋め込み』](#)を参照してください。
- コマンドの発行時にコマンドラインで指定した場合。ほとんどのコマンドでは、`-C` コマンドラインオプションを使用して、コンフィグレーションファイルを指定できます。[『Serena ChangeMan Version Manager コマンドリファレンスガイド』](#)を参照してください。
- `VCSCFG` 環境変数で定義された場合。[『Serena ChangeMan Version Manager コマンドリファレンスガイド』](#)を参照してください。
- 現在の作業ディレクトリ内にローカルコンフィグレーションファイルがある場合。

マスターコンフィグレーションファイル

マスターコンフィグレーションファイルには、ほかのコンフィグレーションファイルによってリセットできないオプションを設定でき、Version Manager の動作を標準化することができます。マスターコンフィグレーションファイル中でオプションを変更禁止に設定すると、Version Manager はマスターの設定に従って動作します。たとえば、マスターコンフィグレーションファイルで新規ファイルのチェックイン時に自動的にアーカイブを作成するオプションを設定し、次にこのオプションを変更禁止に設定すると、このオプションはほかのコンフィグレーションファイルからは変更できず、Version Manager は自動的にアーカイブを作成します。つまり、マスターコンフィグレーション

ファイル中のオプションを設定し、さらにそのオプションを変更禁止に設定すると、ほかのコンフィグレーションファイルではそのオプションを変更することができません。

Version Manager は、default.cfg というデフォルトのマスターコンフィグレーションファイルをインストールします。デフォルトでは、このファイルは Version Manager のインストール時に次の場所に置かれます。

- Windows の場合 : `drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合 : `/usr/serena/vm/common/pvcsprop/pvcs/vm`

このファイルの設定は必要に応じて変更できます。このファイルをコマンドラインインターフェイスのマスターコンフィグレーションファイルとして使用するには、このファイルのコピーを作成し、オプションを必要に合わせて変更し、次にコピーしたファイルを VCONFIG コマンドにより Version Manager に埋め込んでください。Version Manager にコンフィグレーションファイルを埋め込むことにより、すべてのユーザが同一のコンフィグレーション設定を使用し、またユーザが別のマスターコンフィグレーションファイルを使用できないことが保証されます。

Version Manager に埋め込まれたマスターコンフィグレーションファイルは、それが埋め込まれた Version Manager を使用するすべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。マスターコンフィグレーションファイルを埋め込む手順については、[118 ページの『Version Manager へのマスターコンフィグレーションファイルの埋め込み』](#)を参照してください。

コマンドラインインターフェイスでのコンフィグレーションオプション設定のガイドライン

このセクションでは、コマンドラインインターフェイスでのマスターおよびローカルコンフィグレーションファイルに設定する、オプションのタイプに関するガイドラインを示します。

マスターコンフィグレーションファイル

マスターコンフィグレーションファイルは、全体的なオプションを設定し、ほかのコンフィグレーションファイルでは変更できないオプションを指定（禁止に設定）します（前述のマスターコンフィグレーションファイルに関する説明を参照してください）。次にその例を示します。

- Version Manager がユーザを識別するためのログインソース。
- アクセスコントロールデータベースなど、すべてのユーザがアクセスできないプログラムファイルの場所。
- リビジョンへのアクセスと並行開発に影響するロックのオプション。
- 同時アクセスが発生した場合の、Version Manager によるアーカイブの破損防止に影響するセマフォオプション。
- すべてのプロジェクトが使用する必要のあるプロモーションモデル。

ローカルコンフィグレーションファイル

ローカルコンフィグレーションファイルは、ユーザごとのオプションの設定に使用します。ローカルコンフィグレーションファイルでのオプション設定は、マスターコンフィグレーションファイルでの設定について例外を作成する場合に行います。大半のユーザに対するコンフィグレーションオプションと、少数のユーザに対するコンフィグレーションオプションを別々に設定する場合に、ローカルコンフィグレーションファイルを使用することができます。

たとえば、プロジェクトが特定のバージョンラベルに対して作業を行うように設定されているが、1人のユーザが別のバージョンラベルに対して作業を行う必要がある場合は、この例外を定義するため

にローカルコンフィグレーションファイルを使用できます。また、プロジェクトが参照ディレクトリを保持するように設定されていない場合に、個々のユーザがローカルコンピュータ上で変更するファイルの参照コピーを保持できるよう、ローカルコンフィグレーションファイルを定義することができます。

コマンドラインインターフェイスによるコンフィグレーションファイルの読み込み

Version Manager は常に、ローカルコンフィグレーションファイルを読み込む前に、マスターコンフィグレーションファイルを読み込みます。これは、マスターコンフィグレーションファイルには、ほかのコンフィグレーションファイルでは再設定できないオプションが定義されているためです。ローカルコンフィグレーションファイルの設定は、そのオプションがマスターコンフィグレーションファイルによって変更禁止に設定されていないかぎり、マスターコンフィグレーションファイルの設定より優先されます。たとえば、マスターコンフィグレーションファイルでアーカイブへの複数のロックが許可され、このオプションが変更禁止にされていないが、ローカルコンフィグレーションファイルがアーカイブへの複数のロックを許可していない場合は、このローカルコンフィグレーションファイルによるアーカイブのロック設定が使用されます。

Version Manager が、マスターまたはローカルコンフィグレーションファイルを見つけられない場合は、各オプションについてデフォルトの設定が使用されます。オプションのデフォルト設定については、74 ページの『[コンフィグレーションファイルが使用されない場合のデフォルト設定](#)』を参照してください。

ほとんどのコマンドでは、-C コマンドラインオプションを使用して、コンフィグレーションファイルを指定できます。

デスクトップクライアントで作成されたコンフィグレーションファイルのコマンドラインインターフェイスでの使用

デスクトップクライアントで作成と保守が行われるコンフィグレーションファイルは、コマンドラインインターフェイスにも対応しています。コマンドラインインターフェイスではアーカイブの場所を探すために VCSDir ディレクティブを使用する必要がありますが、デスクトップクライアントでは必要ありません。ただし、デスクトップクライアントでも VCSDir ディレクティブは設定できます ([管理] - [詳細設定] - [全般] タブ - 「その他」 - 「アーカイブ検索パス」)。

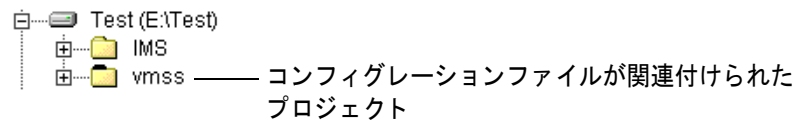
デフォルトでは、デスクトップクライアントでプロジェクトデータベースを作成した場合は、VCSDir がデータベースのアーカイブの場所としてコンフィグレーションファイル内に設定されます。作業ファイルをプロジェクトデータベースに追加し、これらのファイルのアーカイブの場所をプロジェクトデータベースのアーカイブ構造外に設定した場合は、コマンドラインから操作を行えるよう、VCSDir の設定を変更する必要があります。

コンフィグレーションオプションの設定

このセクションでは、デスクトップクライアントおよびコマンドラインインターフェイスでコンフィグレーションオプションを設定する手順について説明し、それぞれのコンフィグレーションオプションを定義します。オプションによっては、このマニュアルのほかの個所で詳細に説明されている場合もあります。その場合は参照先が示されています。

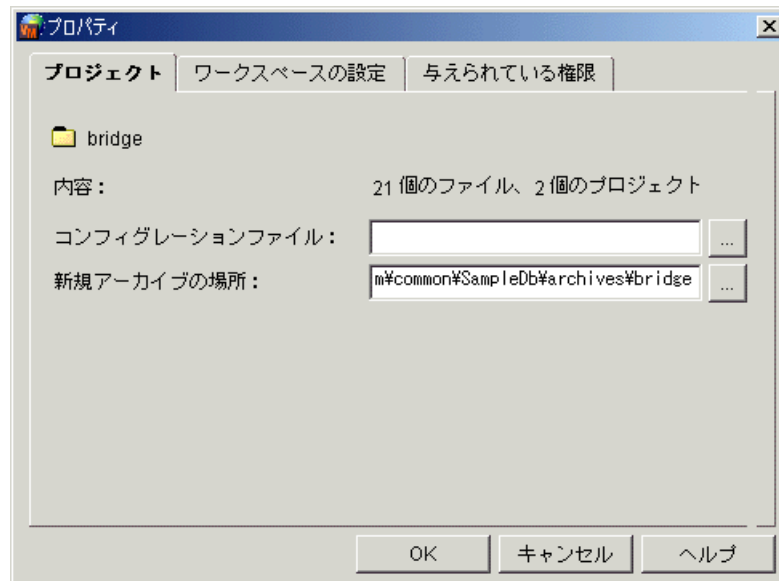
デスクトップクライアントを使用する場合

コンフィグレーションファイルがすでに関連付けられているプロジェクトは、デスクトップクライアントでは黒いタブ付きのフォルダとして表示されます。



プロジェクトデータベースまたはプロジェクト用のコンフィグレーションオプションを設定するには

- 1 設定するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 プロジェクトデータベースまたはプロジェクトにコンフィグレーションファイルが関連付けられている場合は、手順 3 に進みます。それ以外の場合には次の作業を実行してコンフィグレーションファイルを関連付けます。
 - a [ファイル] – [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示され、[プロジェクト] または [プロジェクトデータベース] タブがアクティブになっています。以下は、プロジェクトの [プロパティ] ダイアログボックスです。



- b <コンフィグレーションファイル> フィールドに、プロジェクトに関連付けるコンフィグレーションファイルの場所と名称を入力します。コンフィグレーションファイルがまだ存在していない場合は、ここに指定した場所と名称により自動的に作成されます。
 - c 【OK】 をクリックします。
- 3 [管理] – [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
 - 4 <高度な設定> チェックボックスがまだ選択されていない場合は選択して、選択可能なすべてのオプションを表示します。

- 5 定義するオプションを〔全般〕タブの〈オプション〉リストから選択するか、ほかのタブを選択します。各オプションについては、この章の後のセクションで説明されています。

オプション	参照先ページ
アーカイブの作成	84
ランチ	86
ロック	86
プロモーション	88
アクセスコントロールデータベース	88
ジャーナルファイル	89
ログインソース	90
LDAP の設定	95
作業ファイルの属性	98
キーワード展開	99
参照用ディレクトリ	102
アーカイブ検索パス	104
コマンドライン	106
セマフォ	108
テンポラリファイル	110
ファイルタイプ	112
イベントトリガ	116
許可 / 禁止	116

- 6 オプションの定義を終えたら【OK】をクリックするか、【適用】をクリックして設定を保存し、ほかのオプションの定義を続けます。

コマンドラインインターフェイスを使用する場合

主に Version Manager のコマンドラインインターフェイスを使用する場合は、テキストエディタを使用してコンフィグレーションファイルを編集することもできます。また、デスクトップクライアントを使用して Version Manager を設定し、コンフィグレーションファイルを編集することもできます。デスクトップクライアントによって設定したコンフィグレーションファイルは、コマンドラインインターフェイスでも有効です。

コンフィグレーションファイルを手作業で編集するには、そのフォーマットを理解する必要があります。コンフィグレーションファイルには、ディレクティブを挿入します。ディレクティブとはコンフィグレーションオプションの名称で、たとえば ArchiveSuffix はアーカイブに使用される拡張子を指定するディレクティブです。ディレクティブによっては、ArchiveSuffix = +-arc のように値を指定しなければならないものもあります。ほかのディレクティブ、たとえばアーカイブの更新後にジャーナルファイルに記入を行うよう Version Manager に指示する Journal には、値は必要ありません。

コンフィグレーションファイルには次のルールがあります。

- 通常のテキストファイルでなければならない。
- コメント行はシャープ（#）で開始しなければならない。

- コンフィグレーションファイル中で行を継続するには、行の終端に円記号 (¥) を記入する。この円記号は、行の最後の文字でなければならず、その後にはスペースは付けられません。
- Disallow ディレクティブは、マスターコンフィグレーションファイル内でのみ使用できる。

コンフィグレーションファイルでの条件文の使用

条件に応じて Version Manager にコンフィグレーションファイル中の異なる箇所を読み込ませるため、コンフィグレーションファイル中では条件文を使用できます。

たとえば、システム管理者が時間を節約するため、多大なメンテナンスを必要とするプロジェクトごとのコンフィグレーションファイル設定の代わりに、すべてのプロジェクトに関する設定情報を 1 つのコンフィグレーションファイルに含める場合などが該当します。

詳細については『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』の『コンフィグレーションファイルでの条件文の使用』を参照してください。

アーカイブ作成時のオプション

アーカイブ作成時オプションは、新規に作成されるアーカイブにのみ影響します。これらのオプションを既存のアーカイブに適用するには、デスクトップクライアントでは [管理] - [アーカイブ属性] を選択し、コマンドラインインターフェイスでは VCS コマンドを使用します。

アーカイブ作成時オプションを使用することにより、以下が可能となります。

- 1 つのアーカイブに対する複数ロックの防止。複数のロックを許可すると、アーカイブ中の異なるリビジョンに対し、異なるユーザが同時にロックを設定します。このオプションのディレクティブは ExclusiveLock です。



注 MultiLock と ExclusiveLock ディレクティブは、同時には使用できません。MultiLock ディレクティブは作成されるアーカイブに対しては影響しませんが、ExclusiveLock ディレクティブは影響します。MultiLock については、[86 ページの『ロックオプション』](#)を参照してください。

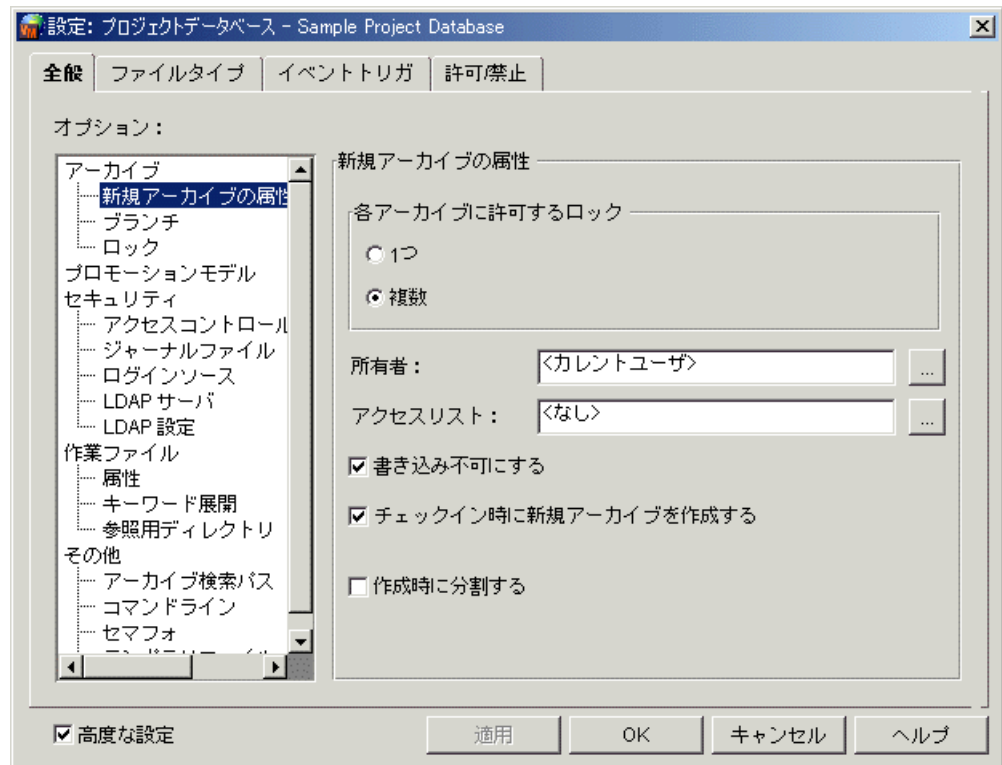
- アーカイブの所有権の設定。このオプションのディレクティブは Owner です。
- アーカイブへのアクセスを許可されたユーザとユーザのグループを特定する、アーカイブのアクセスリストの定義。このオプションのディレクティブは AccessList です。
- アーカイブへの書き込み禁止の設定。これにより、アーカイブを不注意による削除や更新から保護できます。Version Manager のコマンドはアーカイブを更新する前に書き込み禁止を自動的に削除し、後で元に戻します。このオプションのディレクティブは WriteProtect です。
- 作業ファイルをチェックインする時に Version Manager がアーカイブを見つけられない場合の、アーカイブの自動作成。このオプションのディレクティブは AutoCreate です。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 〈オプション〉リストで、「アーカイブ」の下の「新規アーカイブの属性」を選択します。右側に〈新規アーカイブの属性〉表示部が表示されます。各オプションについて表示される値は、プロ

プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



4 オプションを設定するには、次のいずれかの作業を行います。

- 各アーカイブに許可されるロックについて、<1 つ> または <複数> を選択します。
- <所有者> フィールドに、新規アーカイブの所有者のユーザ ID を入力する。値、<現在のユーザ> は、アーカイブが作成されたときに Version Manager を実行していたユーザです。
- <アクセスリスト> フィールドに、新規に作成されたアーカイブにアクセスできるアクセスリストグループおよびユーザ ID を入力する。このフィールドの値は合計で 254 文字を超えることはできません。指定するグループとユーザ ID は、アクセスコントロールデータベースで定義されている必要があります。値はコンマで区切ります。アクセスリストについては、201 ページの『セキュリティの使用』を参照してください。
- <書き込み不可にする> チェックボックスを選択し、アーカイブを書き込み禁止に設定する。このオプションの設定により、アーカイブが不注意による削除や更新から保護されます。
- <新規チェックインの時、アーカイブを自動生成する> チェックボックスを選択する。これで、作業ファイルのチェックイン時に Version Manager がアーカイブを見つけられなかった場合に、自動的にアーカイブが作成されます。選択されていない場合には、アーカイブが作成されないというメッセージが表示されます。このオプションはコマンドラインインターフェイスにのみ適用されます。
- <作成時に分割する> チェックボックスを選択する。これで、新しいアーカイブが Version Manager File Server によってリビジョンパスにマップされている場合に、それらのアーカイブが別々のメタデータとリビジョンコンポーネントに自動的に分割されます。このディレクトティブは、Version Manager File Server のリビジョンライブラリ機能とともに使用できます。詳細については、158 ページの『リビジョンライブラリの作成』を参照してください。

- 5 オプションの定義を終えたら【OK】をクリックするか、【適用】をクリックして設定を保存し、ほかのオプションの定義を続けます。

コマンドラインインターフェイス

アーカイブ作成オプションを定義するディレクティブは次のとおりです。

- ExclusiveLock | NoExclusiveLock。デフォルトは NoExclusiveLock で、各アーカイブに対して複数のロックが許可されます。
- Owner。デフォルトはそのアーカイブを作成したユーザです。
- AccessList。デフォルトではアクセスリストは存在しません。
- WriteProtect | NoWriteProtect。デフォルトではアーカイブは書き込み禁止に設定されます。
- AutoCreate | NoAutoCreate。デフォルトでは、アーカイブが自動的に作成されます。
- RFSSplitOnCreate | NoRFSSplitOnCreate。デフォルトでは、新しいアーカイブは分割されません。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

ブランチオプション

ブランチオプションにより、プロジェクトに自動ブランチを設定できます。ブランチは、開発時に別のラインを設定することで、トランクまたはほかの開発ブランチから 1 つまたはそれ以上のリビジョンが分岐可能です。ブランチにより、トランク上で作業を行っているほかの開発者と平行して、別のバージョンのファイルを開発できます。自動ブランチを設定するためのディレクティブは DefaultVersion、BaseVersion、および BranchVersion の 3 つです。

デフォルトでは、ブランチオプションにはデスクトップクライアントまたはコマンドラインインターフェイスでは値は設定されておらず、自動ブランチは定義されていません。

ブランチに関する詳細な情報とそのセットアップ手順については、[259 ページ](#)の『ファイルのブランチとマージ』を参照してください。

ロックオプション

アーカイブに対するロックに関するオプションでは、次のような設定を選択できます。

- 1 人のユーザに対して複数のリビジョンに対するロック、またはリビジョンごとに複数のロック、またはその両方を許可する。このオプションのディレクティブは MultiLock です。複数のロックに関する詳細については、[266 ページ](#)の『ブランチのための多重ロックの使用』を参照してください。



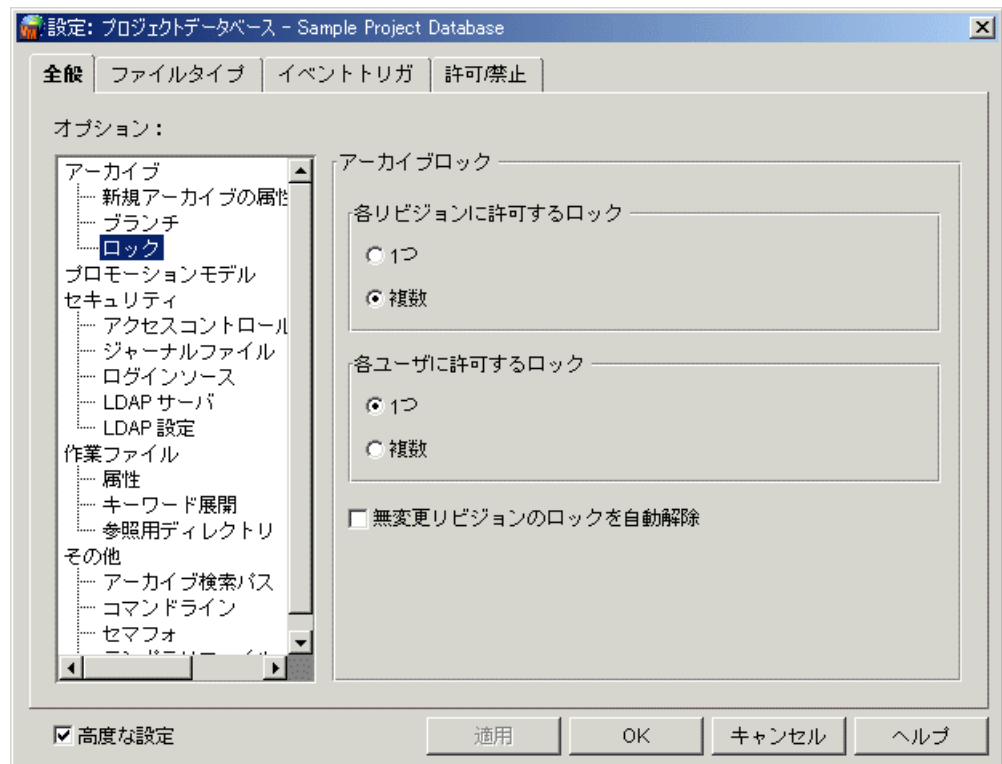
注 MultiLock と ExclusiveLock ディレクティブは、同時には使用できません。MultiLock ディレクティブは作成されるアーカイブに対しては影響しませんが、ExclusiveLock ディレクティブは影響します。ExclusiveLock については、[84 ページ](#)の『アーカイブ作成時のオプション』を参照してください。

- 変更されていないリビジョンのチェックイン時にロックを解除し、チップリビジョン番号を増やさないよう、Version Manager を設定する。このオプションのディレクティブは ForceUnlock です。

デスクトップクライアント

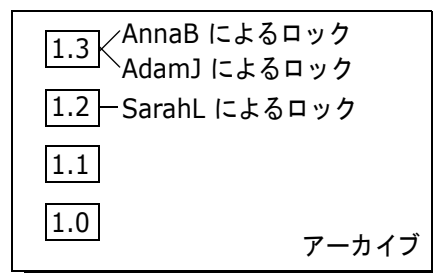
デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 3 <オプション> リストで、「アーカイブ」の下の「ロック」を選択します。右側に<アーカイブのロック> 表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



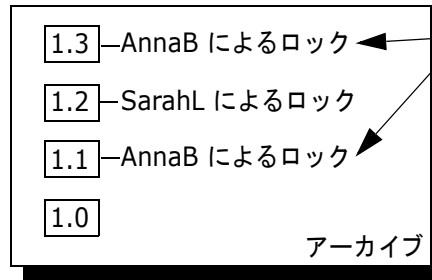
4 オプションを設定するには、次のいずれかの作業を行います。

- 各リビジョンに許可されるロックについて、<1 つ> または <複数> を選択する。単一のリビジョンに複数のロック設定が許可されている場合を次に図示します。



アーカイブ中では各ユーザに許可されるロックは 1 つまでですが、1 つのリビジョンに対する複数のロックは認められます。

- 各ユーザに許可されるロックについて、<1 つ> または <複数> を選択する。ユーザごとに複数のロック設定が許可されている場合を次に図示します。



ユーザーはアーカイブの複数のロックを設定できますが、リビジョンごとに設定できるロックは1つのみです。

- <無変更のリビジョンのロックを自動解除> チェックボックスを選択し、変更されていないリビジョンのチェックイン中にロックを解除し、トップリビジョン番号を増加しないように Version Manager を設定します。このオプションを選択しない場合はチェックインがキャンセルされ、ロックは有効なままとなります。

5 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

アーカイブに対するロック設定を定義するディレクティブは次のとおりです。

- MultiLock | NoMultiLock。デフォルトは NoMultiLock で、複数のユーザーが複数のリビジョンに対してロックを設定できますが、同一のリビジョンに対しては設定できないことを意味します。
- ForceUnlock | NoForceUnlock。デフォルトは NoForceUnlock で、変更されていないリビジョンはチェックインされず、ロックが有効なままとなることを意味します。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

プロモーションオプション

プロモーションオプションでは、各プロモーショングループを次に高いレベルのグループに関連して定義することにより、プロモーションモデルを定義します。このオプションのディレクティブは Promote です。

デフォルトでは、デスクトップクライアントまたはコマンドラインインターフェイスにはプロモーションモデルは定義されていません。

プロモーションモデルの詳細な説明と定義の手順については、[245 ページの『プロモーションモデルの使用』](#)を参照してください。

アクセスコントロールデータベースのオプション

アクセスコントロールデータベースのオプションは、Version Manager がどのアクセスコントロールデータベースを使用するかを指定し、また指定されたアクセスコントロールデータベースを有効にします。これらのオプションのディレクティブは、それぞれ AccessDB と AccessControl | NoAccessControl です。

デフォルトでは、デスクトップクライアントまたはコマンドラインインターフェイスには、使用できるアクセスコントロールデータベースはありません。デスクトップクライアントでは、プロジェクトデータベースの作成時にアクセスコントロールデータベースが作成されますが、不使用に設定されています。

アクセスコントロールデータベースの詳細な説明と指定の手順については、[201 ページの『セキュリティの使用』](#)を参照してください。

ジャーナルファイルのオプション

ジャーナルファイルには、ユーザがアーカイブに対して行う作業が記録されます。ユーザがアーカイブを更新する作業を行うたびに、その作業に関する情報がジャーナルファイルに記録されます。ジャーナルファイルはジャーナルレポートの生成に使用できます。293 ページの『[ジャーナルレポートの生成](#)』を参照してください。

ジャーナルファイルに関しては次の 3 つのオプションが存在します。

- ジャーナルファイルを保持しません。このオプションのディレクティブは NoJournal です。
- アーカイブごとに journal.vcs という名称のジャーナルファイルを保持し、それをアーカイブディレクトリに保存します。このオプションのディレクティブは Journal です。
- プロジェクト中のすべてのアーカイブについて単一のジャーナルファイルを保持し、その名称と保存する場所を指定します。このオプションのディレクティブは Journal = Path\Filename です。

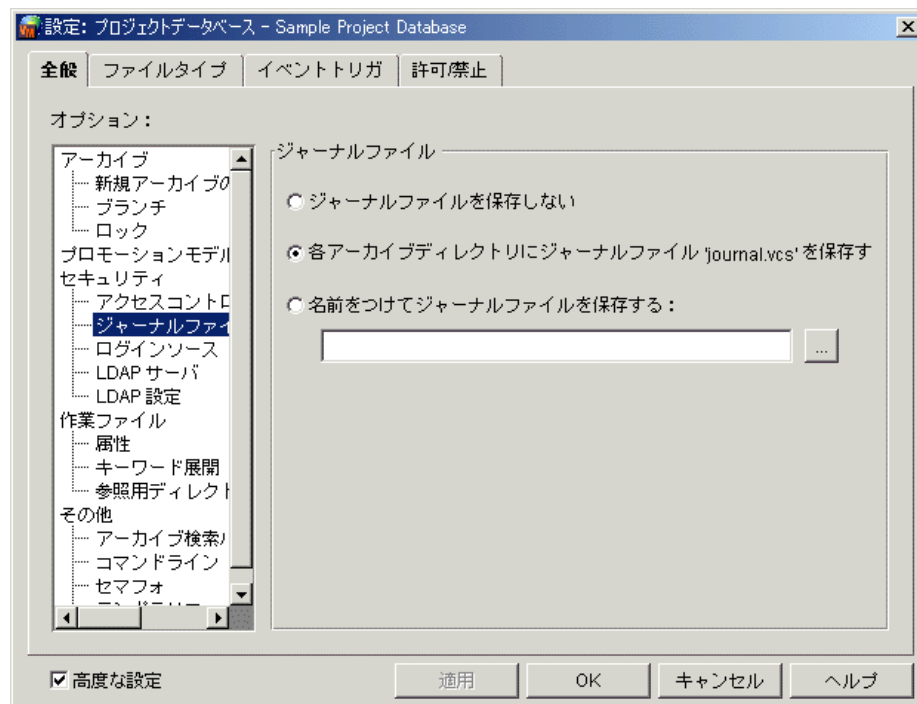


注 パスを指定しない場合、指定した名前のジャーナルファイルが各アーカイブディレクトリに作成されます。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 〈オプション〉リストで、「セキュリティ」の下の「ジャーナルファイル」を選択します。右側に〈ジャーナルファイル〉表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 4 3 つのオプションからいずれかを選択します。

〈名前をつけてジャーナルファイルを保存する〉を選択した場合は、ジャーナルファイルの名称を指定する必要があります。ファイル名とともにパスを指定しない場合は、ジャーナルファイルはアーカイブディレクトリに入れられます。このオプションでは、プロジェクト中のすべてのアーカイブに対して単一のジャーナルファイルが保持されます。

- 5 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス ジャーナルファイルのオプションは、Journal ディレクティブにより設定します。デフォルトではジャーナルファイルは保持されません (NoJournal)。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

ログインソース

ユーザが Version Manager を使用するには、Version Manager がユーザ ID をログインソースから得るように設定する必要があります。ログインソースとは、Version Manager がユーザ ID を取得するために使用するオペレーティングシステム、ネットワーク、またはユーティリティです。Version Manager はログインソースから取得したユーザ ID を、アーカイブ作業の作成者として使用します。

Version Manager は、アクションを実行する前に、[プロジェクトの設定] ダイアログボックスで指定された各ログインソースからユーザ ID を取得しようとします。Version Manager がユーザ ID を取得できない場合は、エラーメッセージを表示して終了します。ユーザ ID が取得されたときに、有効なアクセスコントロールデータベースが存在しない場合は、どのユーザもアーカイブにアクセス可能となります。

有効なログインソースは次のとおりです。

Host ID ホストオペレーティングシステム。このソースは、UNIX や Windows などのユーザ ID を供給するメカニズムを備えたシステム、あるいは複数のネットワークが使用されている環境において使用します。このオプションのディレクティブは LogIn=HOST です。

LDAP ID Lightweight Directory Access Protocol (LDAP)。このソースを使用すると、LDAP サーバに対してユーザ ID とパスワードが認証されます。認証されたユーザ ID はアクセスコントロールデータベースに渡されます (アクセスコントロールデータベースが存在する場合)。アクセスコントロールデータベースでは、パスワードは無視されます。このオプションのディレクティブは LogIn=LDAP です。

LDAP はコマンドラインインターフェイス (CLI) では動作しません。LDAP が指定された最初のログインソースの場合、CLI は次のログインソースを使用します。その他のログインソースが指定されていない場合、CLI コマンドは失敗します。

注： Microsoft Active Directory は LDAP を使用します。

[ログイン] ダイアログボックス

Version Manager デスクトップおよび Web クライアントクライアントのログインユーティリティ。このソースでは、ユーザ ID とパスワードが設定されている場合は、ユーザが Version Manager を使用する前にそれらを入力する必要があります。パスワードを使用するには、アクセスコントロールデータベースの定義が必要です。このログインソースはデスクトップおよび Web クライアントでの操作のみに適用されます。このオプションのディレクティブは LogIn=VLOGIN です。

Netware ID	Novell NetWare (Windows のみ)。このソースは、Windows でなく、Novell NetWare サーバからのユーザ ID の取得に使用します。このオプションのディレクティブは LogIn=NETWARE です。
VCS ID	ユーザの PVCS ID。Version Manager は VCSID 環境変数の値からこれを取得します。このオプションにより使用されるディレクティブは LogIn=VCSID です。 ユーザによる、別のユーザとしてのログインや VCSID 環境変数の値の変更が可能となるため、ユーザ ID のソースとして VCSID を使用することは安全とは言えません。
WNet ID	Microsoft Windows ネットワーク。Version Manager は Microsoft WNET API からユーザ ID を取得します。このオプションのディレクティブは LogIn=WNET です。

Version Manager を実行すると、ログインソースを指定した順にユーザ ID が検索されます。したがって、ログインソースの順番を指定するときには、最も脆弱なシステムが確実に保護されるように、オペレーティングシステムとログインソースのセキュリティ制限を考慮する必要があります。



注 VLOGIN または LDAP がログインソースとして設定されている場合は、[ログイン] ダイアログボックスが表示されます。ここで【キャンセル】をクリックすると、ログイン操作がキャンセルされます。その他のログインソースは使用されず、プロジェクトデータベースにもプロジェクトにもログインできません。

UNIX ユーザの場合 : UNIX デスクトップクライアントと PCLI では、Host ID、LDAP ID、Login Dialog (PCLI ではダイアログではなく、PCLI_ID 環境変数、または -id スイッチを使用します) および VCS ID ログインソースのみがサポートされます。UNIX CLI では、Host ID と VCS ID のみがサポートされます。デフォルト値は Host ID です。

デスクトップクライアントユーザの場合 : Version Manager は、ユーザ ID を取得すると、アクセスコントロールデータベースをチェックしてユーザ ID がそこに存在するかどうかを調べます。Version Manager を設定して、ユーザ ID が存在しない場合に、アクセスコントロールデータベース内にユーザ ID を自動的に作成し、ユーザに権限を割り当てることができます。

これは、アクセスコントロールデータベースを使用するかどうかをまだ決定していない場合に便利です。アクセスコントロールデータベース内にユーザ ID が自動的に作成されるように Version Manager を設定しておけば、後でアクセスコントロールを導入するとき、すでにユーザが定義されているので、セキュリティを設定するための時間を節約できます。また、権限が限定されたゲストアカウントも簡単に設定できます。

ユーザ ID では、大文字と小文字を区別するかどうかを設定することができます。ただし、大文字と小文字の差以外はまったく同じである複数のユーザ ID を定義することはできません。大文字と小文字の区別を制御するディレクティブは CASE です。

Web クライアントユーザの場合 : Version Manager Web クライアントは、次のログインソースと使用可能です。Host ID、LDAP ID、および Login Dialog。これらのログインソースがいずれも使用できない場合、Web クライアントはデフォルトで Login Dialog になります。

シングルサインオン

管理者は、Version Manager または別の Serena アプリケーションにすでにログインしているユーザを認識するよう、Version Manger プロジェクトとプロジェクトデータベースを設定することができます。シングルサインオンが有効になっている場合、ユーザは、Serena License Manager からの有効なライセンスを持っているかぎり、一度ログインするだけで、ログインし直さなくてもシングルサインオンが有効になったすべてのプロジェクトにアクセスできます。シングルサインオンがほか

の Serena アプリケーションでも有効になっている場合、ユーザは、ログインし直さなくても Serena アプリケーション間を切り替えることができます。



注 シングルサインオンを使用するには、Version Manager のユーザ ID と OS のユーザ ID を一致させる必要があります。

トラブルシューティング

シングルサインオンは、FLEXlm ライセンス機能と連動します。シングルサインオンで問題が発生する場合は、ライセンスの設定をチェックしてください。FLEXlm のライセンスについては、『Serena ChangeMan Version Manager インストールガイド』を参照してください。

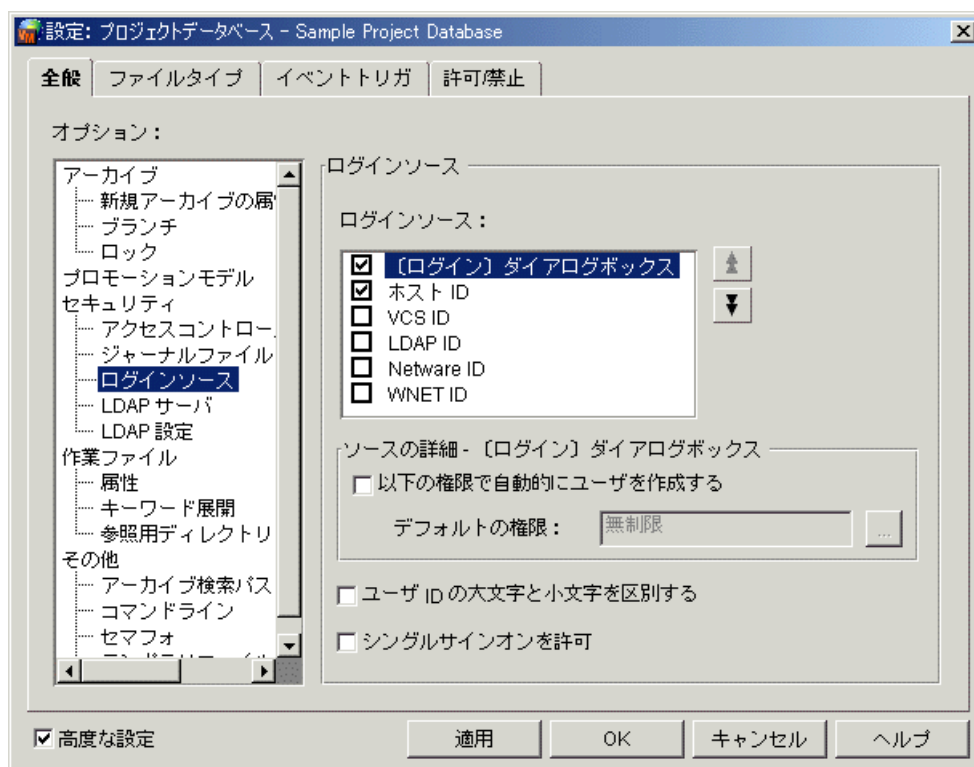
シングルサインオンを有効にするには、次のセクションを参照してください。

ログインソースの設定

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 3 「セキュリティ」の下の「ログインソース」を選択します。[ログインソース] グループボックスが右側に表示されます。



これらのフィールドには、プロジェクトデータベースまたはプロジェクトと関連付けられたコンフィグレーションファイルで定義された現在の設定が表示されます。

- 4 使用するコンフィグレーションに適切なログインソースを選択します。

デフォルトでは、プロジェクトデータベースの作成時に、Host ID がログインソースとしてマスターコンフィグレーションファイル内に作成されます。



- 5 上下の矢印ボタンを使用して、ログインソースを希望の順番に並べます。ここで定義する順番が Version Manager がユーザ ID を探す順番になります。検索は、成功する最初のログインソースで終了します。



注 VLOGIN または LDAP がログインソースとして設定されている場合は、[ログイン] ダイアログボックスが表示されます。ここで【キャンセル】をクリックするかログインに失敗すると、ログイン操作がキャンセルされます。その他のログインソースは使用されず、プロジェクトデータベースにもプロジェクトにもログインできません。

- 6 選択したログインソースを設定して、アクセスコントロールデータベース内にユーザ ID を自動的に作成するには、次の手順に従ってください。
- ログインソースを選択します。
 - 〈以下の権限で自動的にユーザを作成するデフォルト権限〉チェックボックスを選択します。
 - ユーザに割り当てられるデフォルトの権限を指定します。権限の詳細については、[206 ページの『権限について』](#)を参照してください。

デフォルトでは、Host ID ログインソースはアクセスコントロールデータベース内に自動的にユーザを作成し、Unlimited の権限セットを割り当てます。

- 7 ユーザ ID で大文字と小文字が区別されるようにするには〈ユーザ ID で大文字と小文字を区別〉チェックボックスを選択します。デスクトップクライアントのデフォルトでは、ユーザ ID の大文字と小文字は区別されません。
- 8 選択したプロジェクトまたはプロジェクトデータベースへのシングルサインオン形式のアクセスを可能にするには、〈シングルサインオンを許可〉チェックボックスを選択します。詳細については、[91 ページの『シングルサインオン』](#)を参照してください。



注 シングルサインオンを使用するには、Version Manager のユーザ ID と OS のユーザ ID を一致させる必要があります。

- 9 オプションの定義を終えたら【OK】をクリックするか、【適用】をクリックして設定を保存し、ほかのオプションの定義を続けます。

コマンドラインインターフェイス

コマンドラインインターフェイスのオプションを定義するディレクティブは次のとおりです。

- **LogIn**. 有効な値は HOST、LDAP (CLI ではサポートされません)、NETWARE、VCSID、および WNET です。HOST はすべてのプラットフォームに対してデフォルトです。
- **Case**. デフォルトでは、ユーザ ID の大文字と小文字が区別されます。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

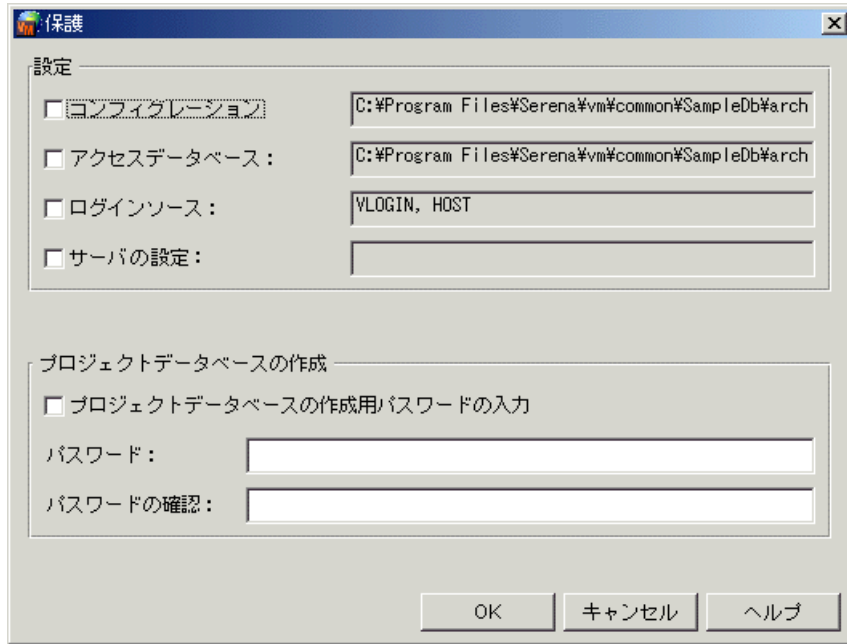
Version Manager へのログインソースの埋め込み

Version Manager にログインソースを埋め込むことにより、すべてのユーザについて同じログインソースを使用してユーザ ID が取得されることが保証されます。ユーザによるログインソースの変更を防止するには、すべてのマスターコンフィグレーションファイルにおいて LogIn ディレクティブを変更禁止に設定し、ユーザには Configure Project 権限を与えないようにします (デスクトップクライアントのみ)。

Version Manager に埋め込まれたログインソースは、そのログインソースが埋め込まれた Version Manager を使用するすべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。

ログインソースを埋め込むには

- 1 該当するログインソースが設定されたマスターコンフィグレーションファイルに関連付けられた、プロジェクトデータベースを選択します。
- 2 [管理] - [設定の保護] を選択します。[設定の保護] ダイアログボックスが表示されます。



- 3 〈ログインソース〉チェックボックスを選択します。このチェックボックスの隣のフィールドに、埋め込まれるログインソースが表示されます。このフィールドは表示専用です。
- 4 【OK】 をクリックします。

ファイル名を埋め込むには、次のように VCONFIG コマンドを使用します。

```
vconfig -isource[,source...] vm_filename
```

ここで、

source は埋め込むログインソースの名称です。有効な値は、HOST、NETWARE、VCSID、WNET、および VLOGIN（デスクトップクライアントのみ）です。

vm_filename は次のいずれかになります。

- Windows : VMWFVC.DLL
- UNIX : vmufvc.a



注 vconfig は、bin ディレクトリのサブディレクトリ、admin にあります。

ログインソースを埋め込んだ後、ファイル (VMWFVC.DLL または vmufvc.a) を Version Manager 実行ファイルと同じ場所に移動する必要があります (まだそこがない場合)。また、この場所に対しては書き込み許可を設定しないよう注意してください。ユーザには Version Manager へのログインソースの埋め込みを許可しないようにします。

VCONFIG コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

LDAP コンフィグレーションオプション

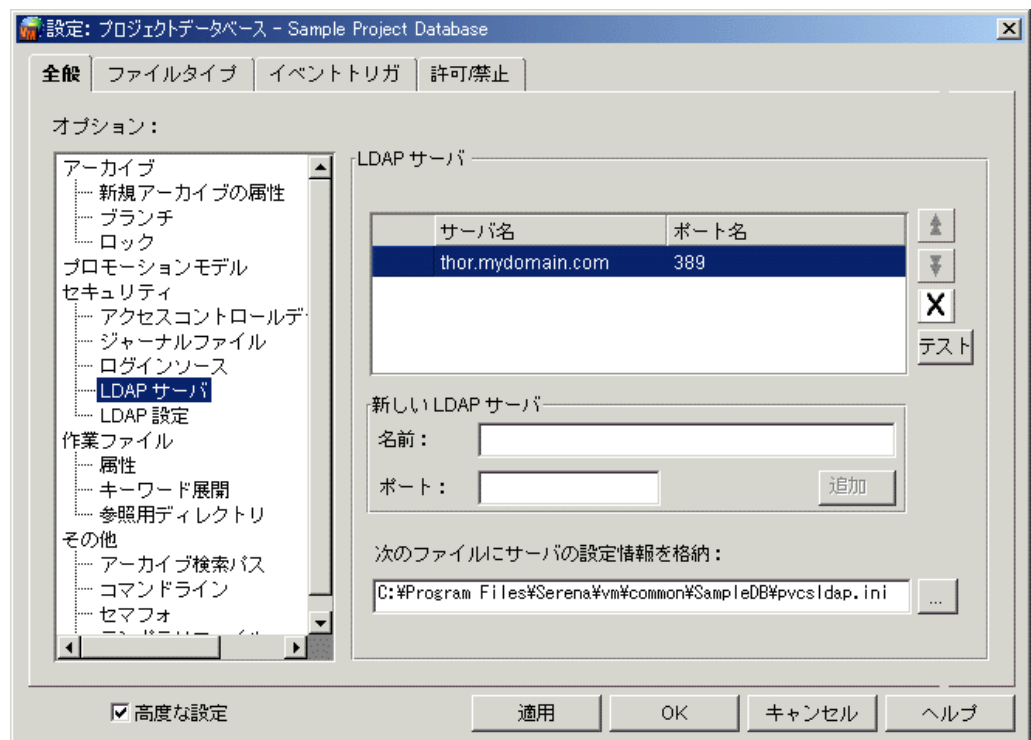
Version Manager を Lightweight Directory Access Protocol (LDAP) サーバと併用するには、以下が必要です。

- 機能する LDAP サーバがある。
- ログインソースの 1 つに「LDAP ID」を選択する。詳しくは 90 ページの『ログインソース』を参照してください。
- LDAP サーバを指定する。詳しくは次のセクションを参照してください。
- LDAP のコンフィグレーションを指定する。詳しくは 96 ページの『LDAP コンフィグレーションの指定』を参照してください。

LDAP サーバの指定

LDAP サーバを指定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 「セキュリティ」の下の「LDAP サーバ」を選択します。[LDAP サーバ] グループボックスが右側に表示されます。



これらのフィールドには、プロジェクトデータベースまたはプロジェクトと関連付けられたコンフィグレーションファイルで定義された現在の設定が表示されます。

- 4 〈LDAP サーバ〉リストでサーバを選択し、以下のいずれかを行います。



- 選択したサーバの〈LDAP サーバ〉リストにおける位置を変更するには、上下の矢印ボタンをクリックする。このリストの一番上から順にログイン認証が自動的に検索されます。



- 選択したサーバをリストから削除するには、【削除】ボタンをクリックする。
- 選択したサーバへの接続をテストするには、【テスト】ボタンをクリックする。

5 〈LDAP サーバ〉リストにサーバを追加するには、以下のフィールドに値を入力します。

- 名前** : LDAP サーバのホスト名または IP アドレスを、`myserver.mydomain.com` のように入力します。
- ポート** : LDAP サーバのポート番号を入力します。通常、LDAP サーバは、SSL の場合にポート 389 と 636 を使用するよう設定されています。
- 追加** : 〈LDAP サーバ〉リストにサーバを追加する際にクリックします。



注 冗長サーバを使用している場合は、各サーバをリストに追加してください。一意のサーバ間の検索に参照を使用している場合、リストで必要となるサーバは 1 つだけです。

6 〈サーバの設定情報を格納するファイルの名前〉フィールドのサーバ設定情報を格納するファイルのパスと名前 (`*.ini`) を入力します。

特記事項

- `ServerConfigInfoPath` ディレクティブが埋め込まれている場合、このフィールドは使用できません。
- この `.ini` ファイルを使用すると、Version Manager および LDAP が有効になっているほかの Serena アプリケーションの間で LDAP 設定情報を共有することができます。LDAP コンフィグレーションファイルを共有するには、ファイルをアクセス可能な場所に置く必要があります。
- デフォルトでは、LDAP コンフィグレーションファイルはプロジェクトデータベースのルートに作られます。
- File Server の管理ユーティリティを使用してクライアント名に LDAP 保護を割り当てた場合、Version Manager File Server は、次のファイルからコンフィグレーション情報を探します。

```
VM_Install\vm\common\bin\OS\pvcsldap.ini
```

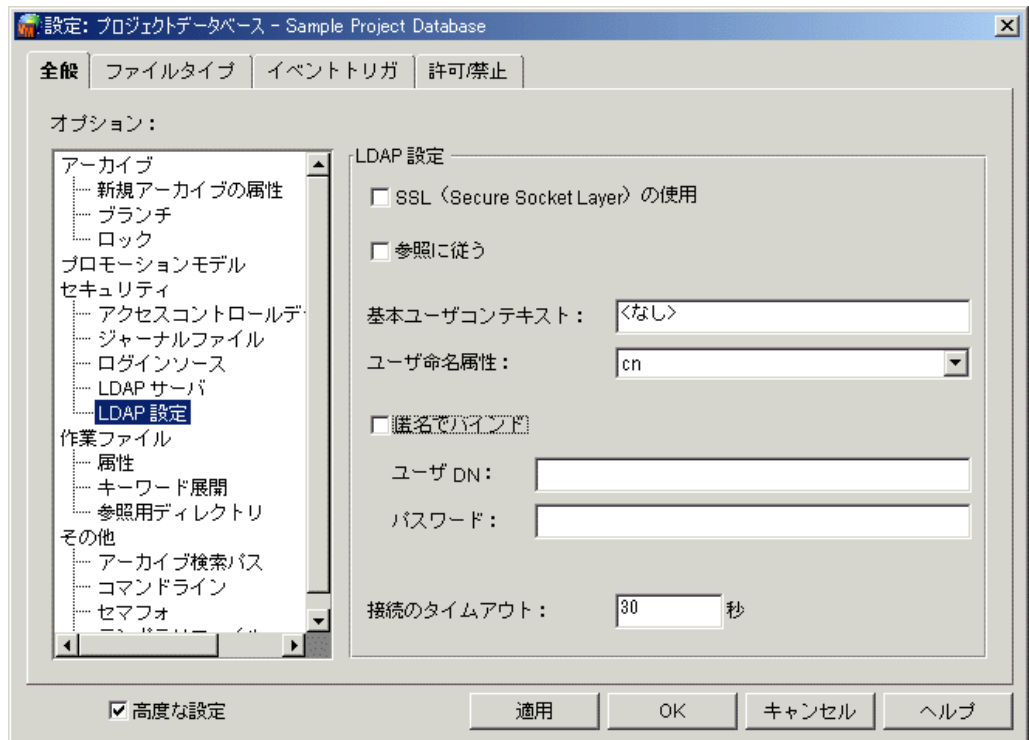
7 【OK】または【適用】をクリックします。

LDAP コンフィグレーションの指定

LDAP コンフィグレーションを指定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。

- 3 「セキュリティ」の下の「LDAP の設定」を選択します。[LDAP の設定] グループボックスが右側に表示されます。



これらのフィールドには、プロジェクトデータベースまたはプロジェクトと関連付けられたコンフィグレーションファイルで定義された現在の設定が表示されます。

- 4 Secure Socket Layer (SSL) を有効にするには、〈SSL の使用〉チェックボックスを選択します。
- 5 Version Manager が LDAP サーバ間で参照をたどれるようにするには、〈参照に従う〉チェックボックスを選択します。この機能は、正しく設定された非冗長の分散サーバで使用してください。
- 6 基本ユーザコンテキスト (識別名) を、〈基本ユーザコンテキスト〉フィールドに指定します。ここからユーザを検索します。
- 7 〈ユーザ命名属性〉リストからユーザ命名属性を選択します。LDAP サーバは、この属性の中にユーザ ID の値を保持します。
- 8 LDAP サーバがユーザリストを取得するときの検索方法を選択します。
- 匿名で検索するには、〈匿名でバインド〉チェックボックスを選択します。LDAP サーバは、匿名ユーザがユーザ、属性リストを取得できるように設定されていなければなりません。



注 Microsoft の Active Directory サーバでは匿名での検索はできません。

- 検索の際のユーザ ID とパスワードの入力を必須にするには
 - a 〈ユーザ DN〉フィールドに、完全なディレクトリユーザ名を指定します。
 - b 〈パスワード〉フィールドに、ユーザパスワードを指定します。

9 〈接続のタイムアウト〉フィールドに、LDAP サーバへの接続がタイムアウトするまでのアイドル状態の時間の長さを秒単位で入力します。

10 【OK】をクリックします。

LDAP のコンフィグレーション例

以下は、一般的な LDAP のコンフィグレーション例を示したものです。

例 1 : Microsoft Active Directory サーバ

サーバ名 : myserver.mydomain.com

サーバポート番号 : 389 (SSL 未使用時は 636)

SSL の使用 : 非選択 (または選択)

基本ユーザコンテキスト : ou=people, dc=mydomain, dc=com

ユーザ命名属性 : sAMAccountName

匿名でバインド : 非選択

ユーザ DN : cn=User One, cn=Users, dc=mydomain, dc=com

パスワード : OpenSesame

例 2 : Netscape Server

サーバ名 : myserver.mydomain.com

サーバポート番号 : 389 (SSL 未使用時は 636)

SSL の使用 : 非選択 (または選択)

基本ユーザコンテキスト : ou=people, dc=mydomain, dc=com

ユーザ命名属性 : uid

匿名でバインド : 選択

ユーザ DN : 無し

パスワード : 無し

作業ファイル属性のオプション

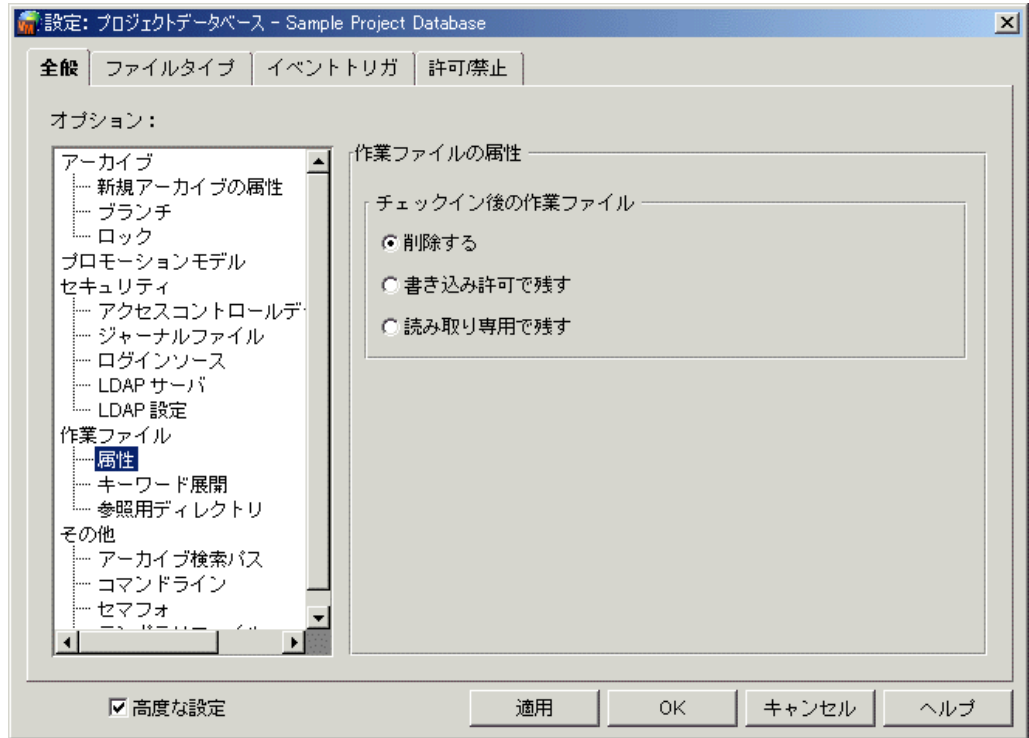
作業ファイルの属性に関し、次のような項目の指定を行うことができます。

- チェックイン後、作業ファイルをその場所から削除するかどうか。チェックイン後に削除されていない作業ファイル中のキーワードは更新されます。このオプションは、作業ファイルをチェックインするときに [オプション] ダイアログボックス ([表示] - [オプション]) で設定を行うことにより変更できます。このオプションのディレクティブは [No]DeleteWork です。
- 作業ファイルのロケーションに残された作業ファイルが書き込み禁止 (読み込み専用) に設定されるかどうか。このオプションのディレクティブは NoDeleteWork=[No]WriteProtect です。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <オプション> リストで、「作業ファイル」の下の「属性」を選択します。右側に<作業ファイルの属性> 表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 3 <チェックイン後の作業ファイル> グループのオプションから、いずれか 1 つを選択します。
 - <削除する> を選択すると、チェックイン後に作業ファイルは削除されます。
 - <書き込み許可で残す> を選択すると、チェックイン後も作業ファイルは削除されず、作業ファイルは書き込み可能モードで残されます。
 - <書き込み禁止で残す> を選択すると、チェックイン後も作業ファイルは削除されず、作業ファイルは書き込み禁止モードに設定されます。
- 4 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

作業ファイル属性は [No]DeleteWork で設定します。デフォルトでは作業ファイルは削除されません。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

キーワード展開オプション

キーワードは、展開されたときにアーカイブに関する情報を定義するよう、作業ファイル中に記述できる特殊な文字列です。たとえば、キーワード、\$Archive\$ はアーカイブの完全なパス名を定義し

まず、作業ファイルがチェックインされたときにキーワードを展開するよう、Version Manager にキーワードオプションを設定することができます。

以下の表は、Version Manager のキーワードの定義の一覧です。キーワードでは、大文字と小文字が区別されます。

キーワード	定義
\$Archive\$	アーカイブの完全なパス名。
\$Author\$	リビジョンの作成者のユーザ ID。
\$Date\$	作業ファイルがチェックインされた日付。
\$Header\$	アーカイブ名、リビジョン番号、リビジョンの日付、および作成者のユーザ ID。
\$Log\$	蓄積されたチェックイン時のメッセージ。
\$Modtime\$	最終更新時刻。
\$Revision\$	リビジョン番号。
\$Workfile\$	Version Manager アーカイブに保存されているファイル名。



注意！ バイナリファイルでキーワードを展開すると、ファイルが破壊されるため、デフォルトでは、特定の拡張子を持つテキストファイルを除き、キーワード展開は無効になっています。バイナリファイルの場合は、キーワード展開を有効にしないでください。

キーワードは、`$Keyword:text$` という形式に展開されます。ここで、`Keyword` はキーワード、`text` はキーワードの現在の値です。たとえば、キーワード、`$Revision$` が展開されたときの値は `$Revision: 1.0$` などとなります。`Log` は展開テキストがドル記号で囲まれていない点でほかのキーワードとは異なります。代わりに、`Log` キーワードを含む行の後に記述の変更が挿入されます。記述の変更は、作業ファイル中で発生と逆の順に蓄積されます。

固定長キーワードの展開

ファイルのタイプによっては、行の長さが固定されていないと正確に読み込めない場合があります。このタイプのファイルには、桁に意味があるファイル、非テキストファイル、ワードプロセッサファイル、データベースファイルおよびスプレッドシートファイルがあります。このように行長を保持するために、キーワードと最後のドル記号 (\$) の間に必要な数の文字を入れ、展開後のキーワードの長さを揃えることができます。以下に例を示します。

```
$Keyword: :$123456: :$
```

各パラメータの意味は次のとおりです。

- **Keyword** はキーワードです。
- **123456** は長さ区切り文字です。長さは文字数で決まります。任意の文字 (:: \$ 以外) を長さ区切り文字に使用することができます。
- **::\$** は、長さ区切り文字の初めと終りを示します。

展開されたキーワードが指定した文字数より少ない場合は、その差がスペースで埋められます。また、指定した文字数より多い場合は、指定した文字数を超える部分は切り捨てられます。



注意！ 固定長キーワード展開の構文が更新されました。古い構文（ここでは省略）も使用することができますが、新しい構文を使用することをお勧めします。古い構文を使用すると、キーワードテキストに文字 \$ が含まれる場合、ファイルが破損する可能性があります。

ディレクティブ

キーワード展開に関して設定可能なオプションは次のとおりです。

- 展開されたキーワードにスラッシュ (/) と円記号 (¥) のどちらのパス区切り文字を使用するか。この区切り文字は、\$Archive\$ などの展開されたキーワードにあるパス指定を区切るために使用されます。このオプションのディレクティブは PathSeparator です。
- チェックイン後にキーワードを展開するときに、作業ファイルのタイムスタンプを更新するかどうか。キーワードを展開しても、チェックイン日時および最終更新日時は影響を受けません。このオプションのディレクティブは ExpandKeywords Touch です。

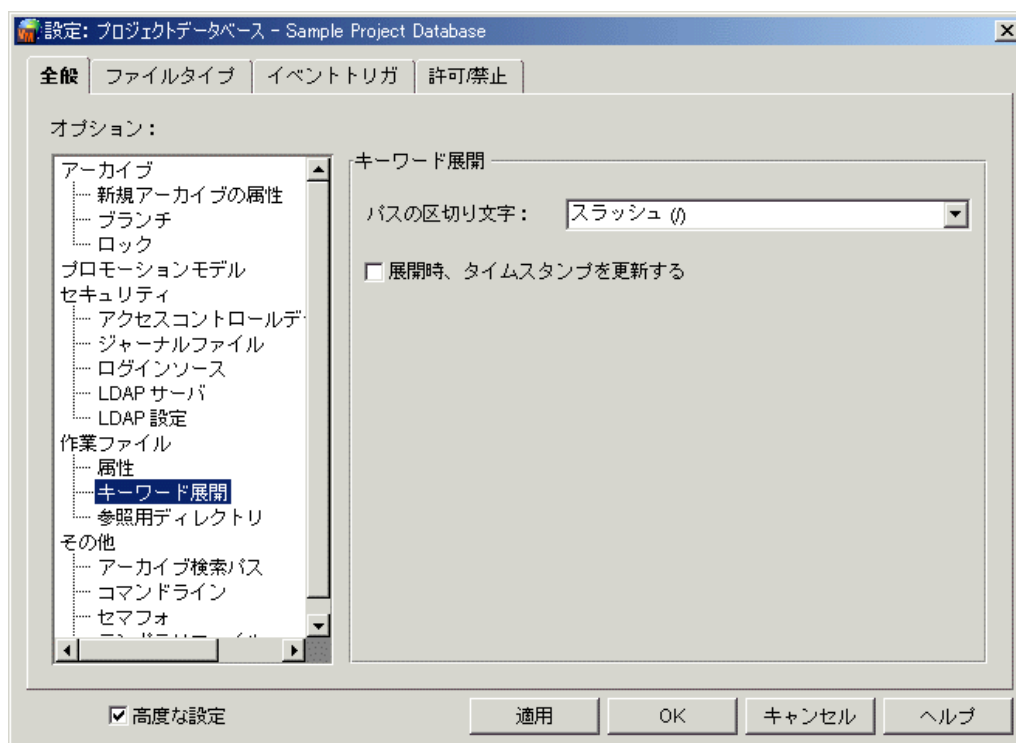
上記以外のキーワード展開のオプションについては、[112 ページの『ファイルタイプに基づいて設定されるオプション』](#)も参照してください。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 3 <オプション> リストで、「作業ファイル」の下の「キーワード展開」を選択します。右側に <キーワード展開> 表示部が表示されます。各オプションについて表示される値は、プロジェク

トデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 4 〈パスの区切り文字〉で「スラッシュ (/)」と「円記号 (¥)」のいずれかを選択します。
- 5 〈展開時、タイムスタンプを更新する〉チェックボックスを選択すると、チェックイン後に Version Manager がキーワードを展開するとき作業ファイルのタイムスタンプを更新します。
- 6 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

コマンドラインインターフェイスのオプションを定義するディレクティブは次のとおりです。

- PathSeparator。デフォルトはスラッシュ (/) です。
- ExpandKeywords Touch。デフォルトではタイムスタンプが更新されます。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

参照用ディレクトリのオプション

参照用ディレクトリは、アーカイブにチェックインするたびに各作業ファイルのコピーが自動的に保存される場所です。参照用ディレクトリは、作業ファイルを参照、印刷、およびコピーするための集中保管場所となります。これらの参照コピーにより、リビジョンをチェックアウトすることなくコードの再使用と検証が可能となります。

書き込み禁止ファイルに対し、読み込みも含めてまったくアクションを実行できないワークベンチまたは統合開発環境を使用しているのではないが、常に参照用ディレクトリ（このセクションで説明）または作業ファイルの場所（98 ページの『作業ファイル属性のオプション』で説明）に作業ファイルの書き込み禁止コピーを置くことをお勧めします。作業ファイルを書き込み禁止にしておく

と、まずアーカイブをロックしてから作業ファイルを更新しなくてはならないことが、すぐにユーザーにわかります。

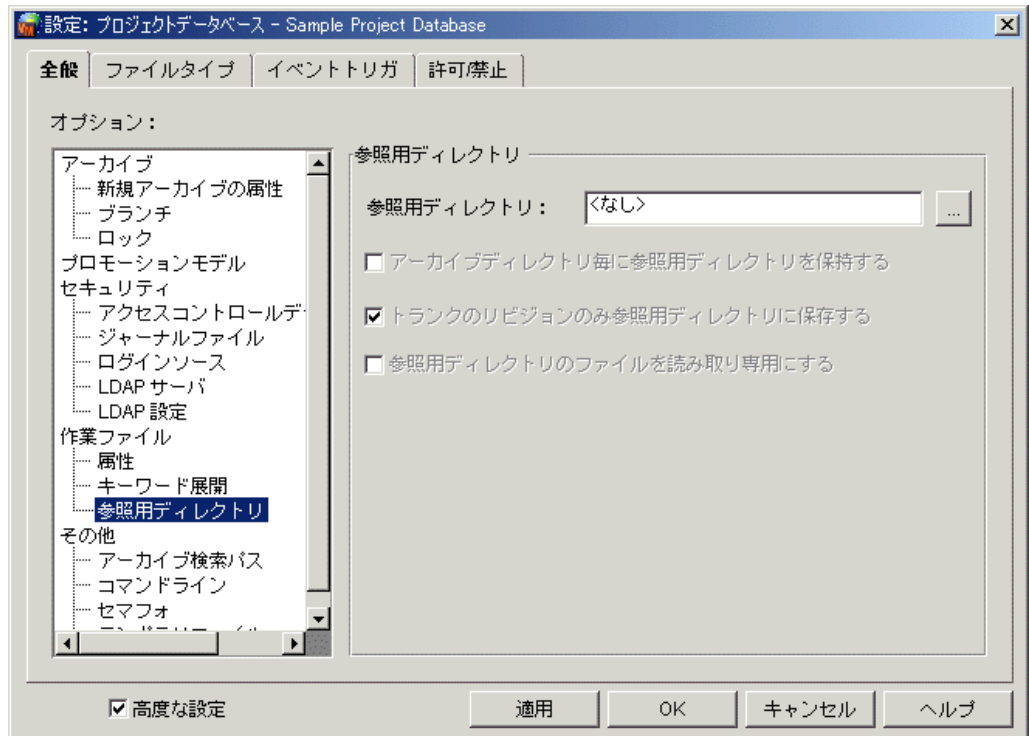
参照用ディレクトリに対して設定可能なオプションは次のとおりです。

- 参照用ディレクトリの場所。このオプションのディレクティブは ReferenceDir です。
- チップリビジョンのコピーについて、トランクのチップリビジョンのみコピーするか、トランクとすべてのブランチの両方のチップリビジョンをコピーするか。このオプションのディレクティブは ReferenceDir=TrunkOnly | All です。
- 参照用ディレクトリ中のファイルを書き込み禁止とするかどうか。このオプションのディレクティブは ReferenceDir=[No]WriteProtect です。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 3 <オプション> リストで、「作業ファイル」の下の「参照用ディレクトリ」を選択します。右側に <参照用ディレクトリ> 表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 4 <参照用ディレクトリ> フィールドに参照用ディレクトリの場所を入力します。

<アーカイブディレクトリごとに参照用ディレクトリを作成> チェックボックスを選択すると、参照用ディレクトリの名称をアーカイブのディレクトリ構造を反映したものにできます。たとえば、d:¥refdir¥project1¥foo.c と d:¥refdir¥project2¥bar.c に対して、c:¥db¥archives¥project1¥foo.c-arc と c:¥db¥archives¥project2¥bar.c-arc を作成します。

- 5 トランクからのチップリビジョンのコピーのみを保持するには、〈トランクリビジョンのみ参照用ディレクトリを保存〉チェックボックスを選択します。
- 6 参照用ディレクトリ中のファイルを書き込み禁止に設定するには、〈読み込み専用の参照用ファイル〉チェックボックスを選択します。
- 7 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

参照用ディレクトリのオプションを定義するディレクティブは ReferenceDir です。デフォルトでは現在の作業ディレクトリに参照用ディレクトリが作成され、参照用ディレクトリ中のファイルが書き込み可能となります。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

アーカイブ検索パスのオプション

アーカイブ検索パスのオプションは、コマンドラインインターフェイスの操作にのみ適用されます。ただし、このオプションの設定は、デスクトップクライアントとコマンドラインインターフェイスのどちらでも行えます。

デスクトップクライアントのアーカイブ検索パスは自動的に定義され、プロジェクトデータベース中で定義または更新されたアーカイブの場所に基づき、自動的に定義されます。

コマンドラインインターフェイスでは、アーカイブ検索パスにより次のものが指定できます。

- ファイルに対して作業を行うときの、アーカイブの検索場所。これによりアーカイブが保存されている場所がわからなくても、ファイルに対する作業を行えます。詳細な情報については『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』の『Version Manager によるアーカイブの検索』を参照してください。このオプションのディレクティブは VCSDir です。
- 指定されたワイルドカードを、最初のアーカイブディレクトリ中のアーカイブのみを探すかどうか。このオプションのディレクティブは FirstMatch です。このディレクティブは、VCSDir ディレクティブと組み合わせて使用されます。VCSDir のみが使用されている場合は、ワイルドカードの指定はすべてのアーカイブディレクトリ中のアーカイブを探すとして解釈されます。
- 作業ファイルの名称のみが指定されている場合に、どのようにアーカイブが検索されるか。このオプションのディレクティブは IgnorePath です。このディレクティブが使用されていると、作業ファイルに対して使用されたパスはすべて無視され、VCSDir 中のアーカイブが検索されます。

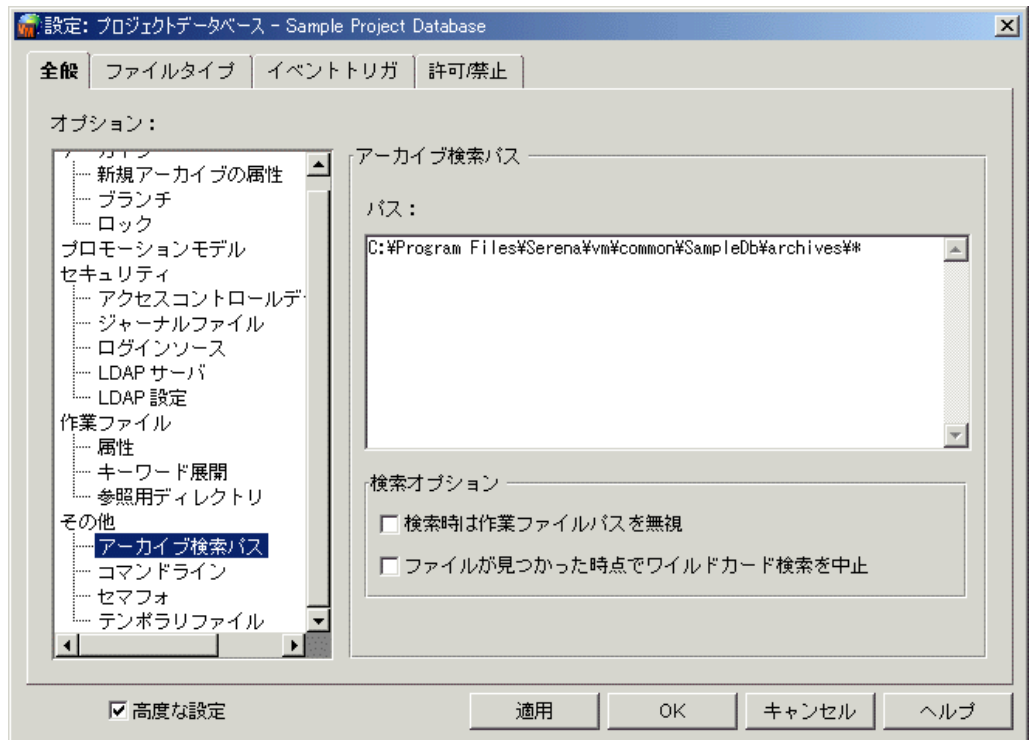
デフォルトでは、デスクトップクライアント内でプロジェクトデータベースを作成するときに、VCSDir がマスターコンフィグレーションファイル中のデータベースのアーカイブの場所に設定されます。カスタマイズにより、サブプロジェクトのためのほかのディレクトリにアーカイブの場所が追加されている場合は、コマンドラインからの操作が機能するよう VCSDir の設定を更新する必要があります。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 〈オプション〉リストで、「その他」の下の「アーカイブ検索パス」を選択します。右側に〈アーカイブ検索パス〉表示部が表示されます。各オプションについて表示される値は、プロジェクト

データベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 4 <パス> フィールドに、アーカイブを検索するディレクトリのリストを入力します。各ディレクトリはセミコロン (;) で区切ります。このディレクトリは、絶対パスと相対パスのどちらを使用することもできます。次に Windows での例を示します。

```
..%vcs; c:%proj%archives
```

リスト中のディレクトリは、左から右に順に検索されます。

- 5 <検索時は作業ファイルを無視> チェックボックスが選択されていると、ユーザにより指定された作業ファイルのパス名は無視され、前述の <パス> フィールドに指定されたディレクトリのみが検索されます。
- 6 <ファイルが見つかった時点で検索を中止> チェックボックスが選択されていると、ワイルドカード検索は最初に一致するものが見つかった時点で終了します。選択されていない場合は、指定されたアーカイブディレクトリがすべて検索されます。
- 7 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

アーカイブ検索パスのオプションを定義するディレクティブは次のとおりです。

- VCSDir. アーカイブを検索する際にディレクトリの指定を可能とします。デフォルトは現在の作業ディレクトリです。
- IgnorePath. 作業ファイルの名称のみが指定された際に、どのように検索が行われるかを指定します。デフォルトは NoIgnorePath です。
- FirstMatch. ワイルドカード検索時に、指定されたディレクトリをすべて検索するか、あるいは最初のマッチングの後に停止するかを指定します。デフォルトではすべてのディレクトリが検索されます。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

コマンドラインオプション

コマンドラインオプションは、コマンドラインインターフェイスの操作にのみ適用されます。ただし、このオプションの設定は、デスクトップクライアントとコマンドラインインターフェイスのどちらでも行えます。

コマンドラインオプションでは、以下を指定することができます。

- 版權とバージョンの情報を表示するかどうか。このオプションのディレクティブは SignOn です。
- 詳細なメッセージを表示するかどうか。このオプションのディレクティブは Quiet | Verbose です。
- 作業ファイルの入力と記述の変更に使用するエディタ。このオプションのディレクティブは VCSEdit です。
- コマンドプロンプトに使用されるテンポラリファイルの名称と場所。このオプションのディレクティブは VCSEdit です。
- メッセージファイルの拡張子。このオプションのディレクティブは MessageSuffix です。
- メッセージファイルを読んだ後に削除するかどうか。このオプションのディレクティブは DeleteMessageFile です。

メッセージファイル

メッセージファイルは、リビジョンのコメントを Version Manager に伝えるために作成するファイルです。メッセージファイルでは、リビジョンのコメントをテキストファイルとして記入できます。これにより、作業ファイルをチェックインするときに、〈コメント〉フィールドに入力する代わりにこのメッセージファイルの内容を使用することができます。

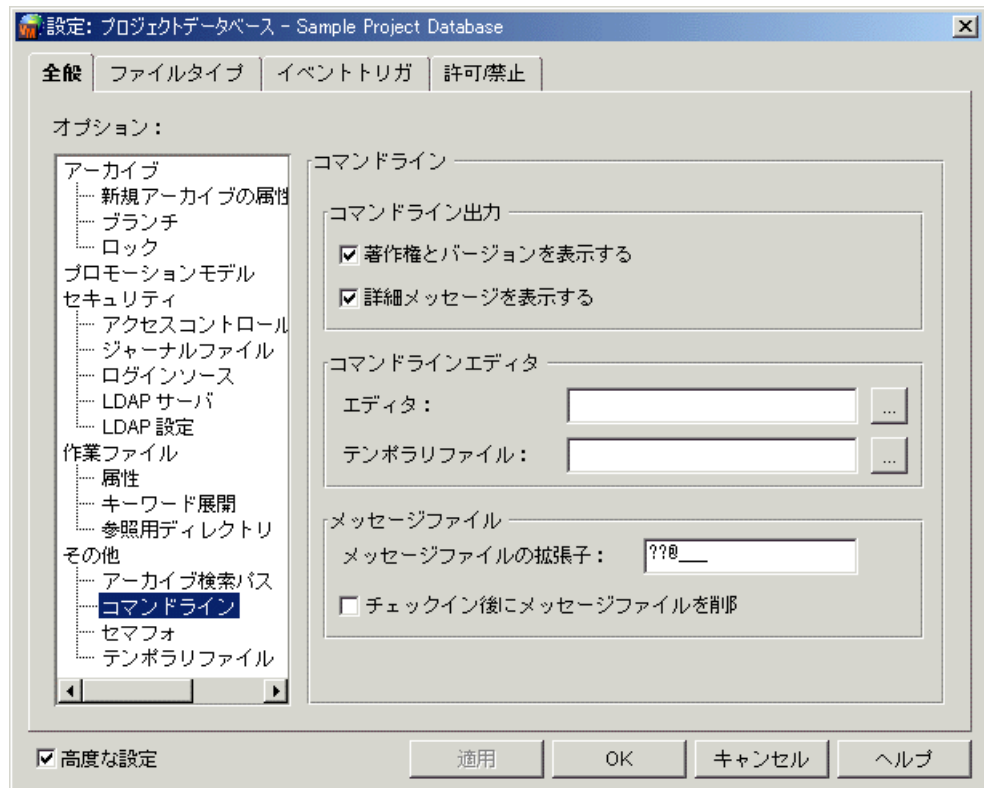
作業ファイルへのチェックイン時にメッセージファイルを使用するには、PUT -M コマンドを使用します。使用するメッセージファイルを指定する (`put -m@message_file`) か、メッセージ拡張子で指定する値を使用して作業ファイルの名称から作成される名称のメッセージファイルを読み込むよう (`put -m@`) に指定できます。後者の場合は、メッセージファイルの拡張子を指定します。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 〈オプション〉リストで、「その他」の下の「コマンドライン」を選択します。右側に〈コマンドライン〉表示部が表示されます。各オプションについて表示される値は、プロジェクトデータ

ベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



4 〈コマンドライン出力〉グループでは、以下を指定します。

- 〈著作権とバージョンを表示〉チェックボックスを選択すると、VCS などの特定の Version Manager コマンドが使用された場合に著作権とバージョンの情報が表示されます。
- 〈詳細情報を表示〉チェックボックスを選択すると、作業ファイルとアーカイブの名称、および現在と以前のバージョン番号など、コマンド操作の詳細が表示されます。

5 〈コマンドラインエディタ〉グループでは、以下を指定します。

- 作業ファイルの入力または記述の変更に使用するテキストエディタの実行ファイルの場所と名称。エディタを指定しない場合は、内部の簡単なラインエディタが使用されます。
- 使用するコマンドラインエディタを指定した場合は、テンポラリファイルの場所と名称を入力する必要があります。テンポラリファイルにはプロンプト行が 2 行コピーされます。これは何についてコメントを記入しているのかを示すためのものです。プロンプト行は削除しないでください。残しておけば、Version Manager によって自動的に削除されます。テンポラリファイルは使用後に削除されます。

6 〈コメントファイル〉グループでは、以下を指定します。

- 〈コメントファイルの拡張子〉フィールドにメッセージファイルの拡張子を入力します。Version Manager デスクトップクライアントのデフォルトでは、拡張子に +-msg が使用されます。たとえば、作業ファイルの名称が windev.c である場合、windev.c-msg というメッセージファイルが検索され、そこから変更コメントが読み出されます。
- 〈チェックイン後にコメントファイルを削除〉チェックボックスを選択すると、変更コメントが読み出された後、メッセージファイルが削除されます。

7 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

コマンドラインのオプションを定義するディレクティブは次のとおりです。

- SignOn。デフォルトでは、Windows では著作権とバージョン情報が表示され、UNIX では表示されません。
- Quiet | Verbose。デフォルトではコマンドライン操作の詳細が表示されます。
- VCSEdit。デフォルトのテキストエディタは Windows ではメモ帳、UNIX では vi です。
- MessageSuffix。コマンドラインインターフェイスのデフォルトの拡張子テンプレートは ??@__ で、作業ファイル名の拡張子の 3 番目の文字がアットマーク (@) で置き換えられます。たとえば作業ファイルの名称が windev.cpp である場合、変更コメントを読み出すために検索されるメッセージファイルの名称は windev.cp@ になります。
- DeleteMessageFile。デフォルトではメッセージファイルは削除されません。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

セマフォオプション

セマフォオプションは、マルチユーザおよびマルチタスクの環境において、プログラムとプロセスによる活動を調整してデータの破損を防ぐのに使用します。このような環境では、セマフォを使用して、共有アーカイブ、ジャーナルファイル、およびアクセスコントロールデータベースに対して Version Manager のプロセスが排他的なアクセスを行えるようにします。セマフォによって、同時にファイルに書き込みを行えるのは 1 つのプロセスのみであることが保証されます。

セマフォはアクセスが開始されたときに作成され、終了したときに削除されます。ほかのユーザやプログラムがそのファイルにアクセスしようとする、そのアーカイブが使用中であることがセマフォによって示されます。

Version Manager には次のタイプのセマフォが存在します。

- ファイルセマフォは、ファイルが使用中であることを示すもので、ファイルが更新される時、および読み出されているときに作成されます。異機種が混在するネットワーク環境では、必ずファイルセマフォを使用してください。UNIX 版の Version Manager では、ファイルセマフォのみがサポートされます。
- **コマンドラインインターフェイスのみ**：Novell NetWare API コールの使用により、NetWare セマフォが作成されます。これはコマンドが異常に終了した場合に、NetWare によって自動的に削除されます。このタイプのセマフォは、更新されるファイルに対してのみ作成されます。



注 コマンドラインインターフェイスのみ：アーカイブのユーザが全員、同じタイプのセマフォを使用するよう、Version Manager を設定します。デスクトップクライアントではファイルセマフォのみが使用可能なため、問題ありません。

セマフォに設定できるオプションは次のとおりです。

- セマフォファイルが作成されるディレクトリ。アーカイブのユーザはセマフォについて、全員が同じディレクトリを使用する必要があります。別々のディレクトリがユーザごとに指定された場合は、セマフォによる保護はすべて失われます。このため、セマフォディレクトリはマスターコンフィグレーションファイル内で設定し、オプションを変更禁止にします。このオプションのディレクティブは SemaphoreDir です。
- Version Manager がアーカイブに対して排他的なアクセスを試みる間隔。このオプションのディレクティブは SemaphoreDelay です。
- Version Manager がアーカイブに対して排他的なアクセスを試みる回数。このオプションのディレクティブは SemaphoreRetry です。

- セマフォファイルの拡張子。このオプションのディレクティブは SemSuffix です。
- **コマンドラインインターフェイスのみ**：ローカルドライブ、および（存在すれば）ネットワークドライブ上に保存されたアーカイブに対して使用するセマフォのタイプ。このオプションのディレクティブは Semaphore です。

コマンドラインインターフェイスでの NetWare セマフォ

NetWare セマフォは更新中のアーカイブに対してのみ作成され、セマフォのチェックは更新中のアーカイブに対してのみ行われます。

アーカイブが更新されないときは、セマフォは作成されません。ほかの更新を伴わないプロセスがアーカイブを読み出す必要があるときは、そのアーカイブがファイル共有モードで開かれます。

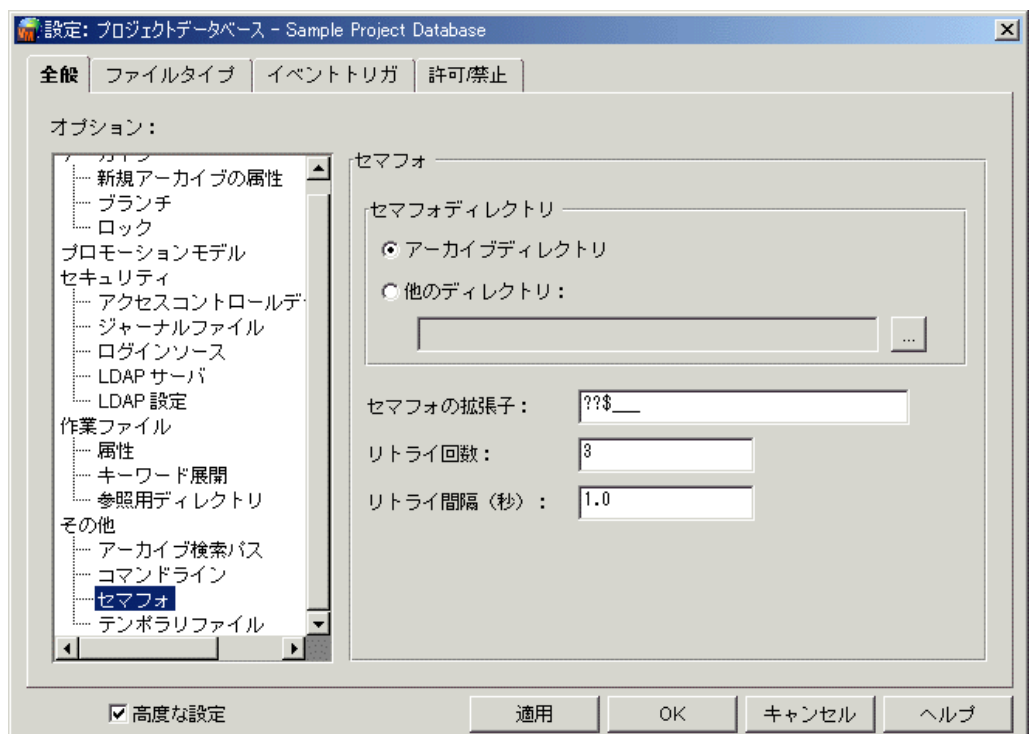
アーカイブの更新中にはセマフォが作成され、そのアーカイブは書き込み禁止モードで開かれます。このアーカイブはアーカイブのテンポラリファイルディレクトリにコピーされ、アーカイブ自体は閉じられます。この段階で、アーカイブを更新しているコマンドには、そのアーカイブの専用のコピーが得られます。セマフォの存在は、ほかのすべての Version Manager コマンドに対し、更新が進行中であり、ほかの更新操作を行えないことを示します。

アーカイブの専用コピーへの更新が終了すると、オリジナルのアーカイブはその専用コピーによって置き換えられます。そのときにほかのプロセスがそのアーカイブを読み出している場合は、アーカイブを更新しようとしているプロセスに対して共有違反エラーが戻されます。セマフォに対してリトライオプションを設定している場合は、更新プロセスはアーカイブが利用できるようになるまで待ち、それから更新を行います。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 <オプション> リストで、「その他」の下の「セマフォ」を選択します。右側に <セマフォ> 表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 3 <セマフォディレクトリ>グループから、次のいずれかを選択します。
 - <アーカイブディレクトリ>。選択すると、セマフォファイルは各アーカイブディレクトリに置かれます。
 - <ほかのディレクトリ>。選択すると、アーカイブディレクトリ以外の、セマフォファイルを置くディレクトリを指定することができます。
- 4 <セマフォの拡張子>フィールドに、セマフォファイルの拡張子を指定します。デフォルトでは +-sem が使用されます。たとえば、アーカイブのファイル名が windev.c-arc である場合、セマフォファイルの名称は windev.c-arc-sem になります。
- 5 <リトライ回数>フィールドに、ネットワーク環境のアーカイブまたはジャーナルファイルに対し、Version Manager が排他的なアクセスを試みる回数を指定します。
- 6 <リトライ間隔 (秒)>フィールドに、アーカイブ、ジャーナルファイル、およびアクセスコントロールデータベースに対して排他的なアクセスを試みる間隔を秒単位で指定します。
- 7 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

セマフォのオプションを定義するディレクティブは次のとおりです。

- Semaphore。デフォルトではセマフォが使用されます。
- SemaphoreDelay。アーカイブ、ジャーナルファイル、およびアクセスコントロールデータベースに対して排他的なアクセスを試みる間隔を指定し、Semaphore ディレクティブと組み合わせ使用されます。デフォルトは 10 (1 秒) です。
- SemaphoreDir。デフォルトではセマフォファイルはアーカイブディレクトリに作成されます。
- SemaphoreRetry。デフォルトは 3 回です。
- SemSuffix。コマンドラインインターフェイスのデフォルトの拡張子テンプレートは ??\$_ で、ファイル名拡張子の 3 番目の文字がドル記号 (\$) で置き換えられます。たとえばアーカイブファイルの名称が windev.c_v である場合、セマフォのファイル名は windev.c_\$ になります。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

テンポラリファイルのオプション

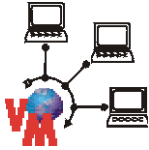
テンポラリファイルは、Version Manager によって作成と削除が行われるバックアップ用ファイルで、アーカイブとその内部データを保護するために使用されます。たとえば、ネットワークに保存されたアーカイブからファイルをチェックアウトする間にネットワークが停止し、このネットワークのクラッシュによりアーカイブが破損した場合に、ネットワークが回復した後で破損していないアーカイブのコピーをテンポラリファイルから取り出すことができます。

Version Manager では次の 2 つのタイプのテンポラリファイルが作成されます。

- デルタ処理やその他のアーカイブ操作に関する情報を含むもの。デフォルトではデルタ処理に関するテンポラリファイルは現在の作業ディレクトリに置かれます。テンポラリファイルに設定される名称は PVnnnnnn.TMP で、nnnnnn はプログラムが指定する文字列です。テンポラリファイルの名称は指定できません。
- テンポラリアーカイブファイルを含むもの。デフォルトではテンポラリアーカイブファイルは現在の作業ファイルディレクトリに作成されます。テンポラリファイルに設定される名称は PVCSnnnn.TMP で、nnnn は 16 進数です。テンポラリファイルの名称は指定できません。テンポラリアーカイブファイルは、ほかのディレクトリを指定して保存することもできます。

テンポラリファイルのオプションでは、次の項目について指定できます。

- デルタ処理のときに生成されるテンポラリファイルの保存場所。このオプションのディレクトリは WorkDir です。
- アーカイブを更新する前に、その一時的なコピーを保存する場所。このオプションのディレクトリは ArchiveWork です。

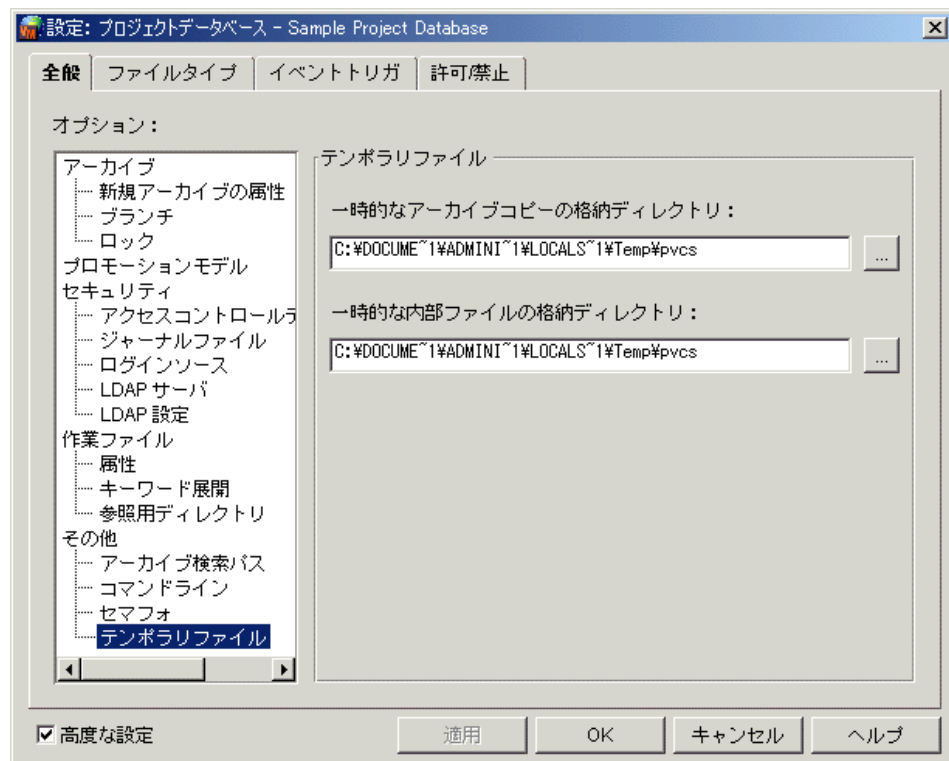


デスクトップクライアント

重要！ Version Manager File Server を使用している場合は、既存の OS のドライブマッピングによってアクセス可能な場所でないかぎり、WorkDir または ArchiveWork を File Server にマップされたディレクトリに設定しないでください。File Server 自体は、ディレクトリのアクセスに使用することはできません。

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 2 〈オプション〉リストで、「その他」の下の「テンポラリファイル」を選択します。右側に〈テンポラリファイル〉表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 3 テンポラリアーカイブファイルを保存するディレクトリを入力します。
- 4 テンポラリな内部ファイルを保存するディレクトリを入力します。
- 5 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

テンポラリファイルのオプションを定義するディレクティブは次のとおりです。

- WorkDir。デルタ処理やほかのアーカイブファイル操作の際に作成されるテンポラリファイルについて、ディレクトリの指定を行います。このディレクトリはテンポラリのバックアップアーカイブファイルが保存される場所ではありません。デフォルトは現在の作業ディレクトリです。
- ArchiveWork。テンポラリアーカイブファイルの作成とその保存先ディレクトリを指定します。デフォルトでは、テンポラリアーカイブファイルが作成され、現在の作業ディレクトリに保存されます。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

ファイルタイプに基づいて設定されるオプション

[プロジェクトの詳細設定] ダイアログボックスの [ファイルタイプ] タブからは、ファイルごとのタイプに基づいて次のオプションを設定できます。

- アーカイブ拡張子。アーカイブ拡張子については、[112 ページの『アーカイブ拡張子の変更』](#)を参照してください。このオプションのディレクティブは ArchiveSuffix です。
- デルタを保存するかどうか。[113 ページの『デルタの保存』](#)を参照してください。このオプションのディレクティブは GenerateDelta です。
- EOL シーケンスを変換するかどうか。このオプションのディレクティブは Translate です。
- キーワードを展開するかどうか。キーワードを展開する場合は、\$Log\$ キーワード展開の行の前に挿入するコメント接頭辞と、\$Log\$ キーワード展開の EOL シーケンスのインジケータの 2 つのオプションが使用できます。これらのオプションのディレクティブは、ExpandKeywords、CommentPrefix、および NewLine です。\$Log\$ キーワードについては、[99 ページの『キーワード展開オプション』](#)を参照してください。



注意！ キーワードの展開はバイナリファイルを破壊するため、Version Manager のデフォルトでは、特定の拡張子を持つテキストファイル以外のキーワードは展開できないよう設定されています。バイナリファイルについてはキーワード展開を使用しないようにしてください。

- カラムマスクとリナンバ。これについては、[113 ページの『カラムマスクとリナンバ』](#)で説明しています。このオプションのディレクティブは、ColumnMask と Renumber です。
- 論理的行数に基づいてデルタを生成するための固定長ファイルのレコードの長さ。これにより固定長のレコードを処理するときのパフォーマンスが向上します。可変長のファイルについてはこのオプションは使用しないでください。このオプションのディレクティブは RecordLength です。

アーカイブ拡張子の変更

デスクトップクライアントのデフォルトでは、アーカイブの拡張子は作業ファイルの名称の最後に -arc を追加したもの（たとえば myfile.txt-arc）で、これにより互いに識別可能なアーカイブ名が作成されます。コマンドラインインターフェイスのデフォルトでは、作業ファイル名の拡張子の 3 番目の文字が V で置き換えられ（たとえば myfile.txv）同一のアーカイブ名が作成される場合があります。たとえば、コマンドラインインターフェイスのデフォルト拡張子を使用することにより、myfile.cpp と myfile.cpy という名称の 2 つの作業ファイルからは、いずれも myfile.cpv という名称が作成されます。

アーカイブ拡張子の定義には 2 とおりの方法があります。1 番目の方法では、作業ファイル名の後に文字が追加されます。この場合は、アーカイブ拡張子をプラス記号 (+) で始めます。このプラス記号以降の文字は作業ファイル名の後に追加されます。たとえば、アーカイブ拡張子が +-arc で作業ファイルの名称が main.c である場合、アーカイブの名称は main.c-arc になります。

2 番目の方法では、たとえば `??V__` のように、拡張子マスクとデフォルト拡張子の 2 つの部分から成るアーカイブ拡張子を定義できます。最初の 3 文字は拡張子マスクで、作業ファイルの拡張子が 3 文字で構成されている場合に、どの文字をアーカイブ拡張子に使用するかを決定します。たとえば、デフォルトの拡張子マスクにある疑問符は、作業ファイル拡張子が存在する場合にその 1 番目と 2 番目の文字を使用することを指示します。文字 `V` は、作業ファイル拡張子が存在する場合に、その 3 番目の文字を `V` で置き換えることを示します。チェックインされた作業ファイルが `main.asm` である場合は、作成されるアーカイブは `main.asv` になります。

2 番目の 3 文字は拡張子のデフォルトで、作業ファイルの拡張子に文字が存在しない場合に、拡張子として使用する文字を指定します。たとえば、デフォルト拡張子のアンダースコア (`_`) は、作業ファイルの拡張子に文字が存在しない場合にアンダースコアで代用することを示します。チェックインされた作業ファイルが `main.c` である場合は、作成されるアーカイブは `main.c_v` になります。

`??V__` によるアーカイブ拡張子の定義は、拡張子中の 3 番目の文字が `V` である作業ファイルでは、作業ファイルとアーカイブの名称が同じになってしまう（たとえば、作業ファイル、`test.java` のアーカイブの名称は `test.java`）ため使用できません。

デルタの保存

デルタとは変更箇所を意味し、これにより各リビジョンが定義され、結果としてアーカイブのサイズが小さくなります。デフォルトでは、Version Manager がファイルを（拡張子を判別して）テキストファイルと認識したら、そのファイルのチップリビジョンは完全に保存され、チップリビジョンではないものはデルタとして保存されます。

作業ファイルがバイナリフォーマットの場合は、デルタのセットは作業ファイル自体と同程度の大きさになることが多く、デルタの計算に時間がかかります。この場合には、作業ファイルの完全なコピーを保存した方がよい場合があります。デフォルトではバイナリファイルについてはデルタは保存されません。

カラムマスクとリナンバ

作成と更新を行うファイルがたとえば COBOL ソースコードファイルのように行番号を含む場合は、カラム（桁）マスクを行う必要があります。カラムマスクにより、チェックインと同時にスペースに変換される桁、およびファイルの比較またはマージ時にスペースとして処理される桁が指定されます。このオプションのディレクティブは `ColumnMask` です。このディレクティブはアーカイブの作成時にのみ影響します。

COBOL ファイルに対して行の追加または削除を行う場合は、ファイルの行番号は自動的に再設定されます。デルタまたは相違点レポートの生成時には、行ごとの比較が行われます。

マスクを行わない場合は、行番号が異なるため、新しいファイルのすべての行が変更されたものと見なされます。これによりデルタと相違点レポートのサイズが大きくなります。

リビジョンをチェックアウトするときにも行番号を挿入するよう、リナンバオプションを設定できます。Version Manager は、桁 1 からはじまるマスクされた桁のみをリナンバします。行番号は左側に 0 が記入されます（000010、000020 等）。このオプションのディレクティブは `Renumber` です。



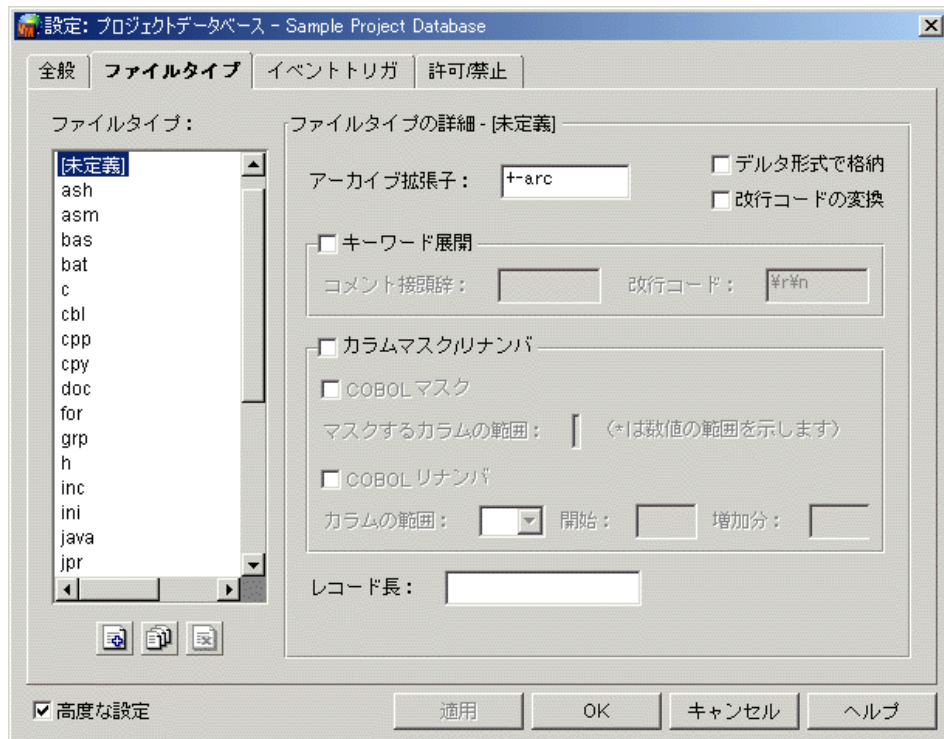
注 レコードの終端を超える桁を指定した場合、行にスペースは挿入されません。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。

- 2 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 3 [ファイルタイプ] タブを選択します。右側に〈参照用ディレクトリ〉表示部が表示されます。各オプションについて表示される値は、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中で現在定義されている設定です。



- 4 左側の〈ファイルタイプ〉リストから、オプションを定義するファイルタイプを選択します。[未定義] はデフォルトのファイルタイプで、特に指定のないすべてのファイルタイプです。[未定義] ファイルタイプの設定はほかのタイプに変更することができます。

- 5 次のいずれかの操作を行い、ファイルタイプリストを管理します。



- 完全に新しいファイルタイプを追加します。新規のファイルタイプは [未定義] ファイルタイプで定義されたデフォルトの設定を継承します。



- 既存のファイルタイプを複製することにより追加します。新規のファイルタイプは複製元のファイルタイプの設定を継承します。



- ファイルタイプを削除します。



注 Version Manager からは、.c、.h、.pas、.mak、.for、.bas、.prg、.asm、.bat、.obj、.lib、.doc、.y、.l、.o、および .a のファイルタイプは削除できません。デスクトップクライアントではリストから削除できますが、次回にこのダイアログボックスを開いたときに、これらのファイルタイプが再度表示されます。

- 6 〈アーカイブ拡張子〉フィールドにアーカイブの拡張子を入力します。112 ページの『アーカイブ拡張子の変更』を参照してください。
- 7 チップリビジョン以外のすべてのリビジョンをデルタのセットとして保存するには、〈デルタ形式で格納〉チェックボックスを選択します。113 ページの『デルタの保存』を参照してください。

- 8 EOL シーケンスを変換するには〈改行コードの変換〉チェックボックスを選択します。テキストファイルについては、常にこのオプションを選択する必要があります。



注意！ バイナリファイルではこのオプションは選択しないでください。アーカイブが壊れることがあります。

- 9 作業ファイルをチェックインするときにキーワードを展開するには、〈キーワード展開〉チェックボックスを選択します。キーワードの説明については、99 ページの『キーワード展開オプション』を参照してください。



注意！ バイナリファイルではこのオプションは選択しないでください。アーカイブが壊れることがあります。

このオプションを選択すると、以下を実行することができます。

- \$Log\$ キーワードのコメント接頭辞を定義するには、〈コメントの接頭辞〉に値を入力します。コメント接頭辞はコメント行を見やすくするためのものであり、ソースファイルのコメント中への \$Log\$ キーワードの挿入は、開発者が行う必要があります。
- 〈改行コード〉フィールドに値を入力して、\$Log\$ キーワードの改行コードを定義します。デフォルトでは、\$Log\$ キーワードのために追加された行の終端には、キャレットリターン (CR) と行送り (LF) の組み合わせが挿入されます。使用するオペレーティングシステムがこれ以外の EOL シーケンスを必要とする場合には、それをここで指定します。

- 10 作成と変更を行うファイルに行番号が使用されている場合は、〈カラムマスク / リナンバ〉チェックボックスを選択します。この場合は次のような作業を行えます。

- COBOL ファイルに対するデフォルトの COBOL マスク定義を使用するには、〈COBOL マスク〉チェックボックスを選択します。これは 73-80 桁をマスクし、第 1 桁目が数値の場合にのみ、1-6 桁をマスクします。
- 〈COBOL マスク〉を選択しない場合は、マスクする最初と最後の桁を入力します。たとえば、1 と 6 を入力すると、第 1 桁から第 6 桁目までが無視されます。マスクングを数値によるフィールドのみに限定するには、範囲の隣にアスタリスク (*) を入力します。たとえば、1-6* を指定すると、第 1 桁目が数値の場合にのみマスクングが行われます。また、1-6*,10-12 のように複数の範囲を入力することもできます。それぞれの範囲はコマで区切ります。
- COBOL ファイルに対するデフォルトの行番号再設定の定義を使用するには、〈COBOL リナンバ〉チェックボックスを選択します。これにより、第 1 桁から第 6 桁目までが、10 から開始して 10 ずつ増加する番号で再設定されます。
- 〈COBOL リナンバ〉を選択しない場合は、次のように入力します。
 - 番号の再設定を行う桁の範囲を〈カラムの範囲〉フィールドに入力。たとえば、第 1 桁から第 6 桁目までに対して番号を再設定するには 1-6 と入力します。
 - 〈開始カラム〉フィールドに最初の行番号を入力。
 - 〈増加分〉フィールドに行番号の増加分を入力します。たとえば、最初の番号に 20 を入力し、増加分を 5 と入力した場合は、最初の行の番号は 000020、次の行は 000025、3 番目は 000030 のようになります。ここでは 6 桁の番号を再設定するものと仮定しています。

カラムマスクとリナンバの説明は、113 ページの『カラムマスクとリナンバ』を参照してください。

- 11 固定長のレコードを持つファイルでは、〈レコード長〉フィールドにレコードの長さを入力することにより、論理行に基づいてデルタが生成されます。



注意！ 可変長のファイルについてはこのオプションは使用しないでください。使用すると、アーカイブが破損したり、ほかの予想されない結果が生じることがあります。

- 12 オプションの定義が終了したら【OK】をクリックします。または【適用】をクリックして設定を保存して、さらにオプションを定義します。

コマンドラインインターフェイス

ファイルタイプのオプションをサポートするディレクティブは次のとおりです。

- ArchiveSuffix。デフォルト値は ??V___ です。
- GenerateDelta。デフォルトではデルタが生成されます。
- Translate。デフォルトでは EOL シーケンスは変換されません。
- ColumnMask。デフォルトでは COBOL ファイルについて第 1 桁から第 6 桁目までがマスクされます。このディレクティブは作成されるアーカイブのみに影響し、既存のアーカイブには影響しません。既存のアーカイブに関してカラムマスクを変更するには、VCS コマンドを使用します。
- Renumber。デフォルトではリナンバは行われません。このディレクティブは作成されるアーカイブのみに影響し、既存のアーカイブには影響しません。既存のアーカイブに関してリナンバオプションを変更するには、VCS-XRenumber コマンドを使用します。
- RecordLength。デフォルトのレコード長はありません。

これらのディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

イベントトリガ

[イベントトリガ] タブでは、指定した Version Manager イベントが発生する前、または発生した後に実行するプログラムを指定できます。イベントとは Version Manager による処理中に発生する特定のアクションを意味し、たとえばファイルのチェックイン、バージョンラベルの割り当て、およびジャーナルファイルへの項目追加などが含まれます。このオプションのディレクティブは EventTrigger です。

デフォルトでは、デスクトップクライアントまたはコマンドラインインターフェイスにはイベントトリガは定義されていません。

イベントトリガとそのセットアップに関する詳細な説明は、271 ページの『イベントトリガの使用』を参照してください。

オプションの変更許可と変更禁止

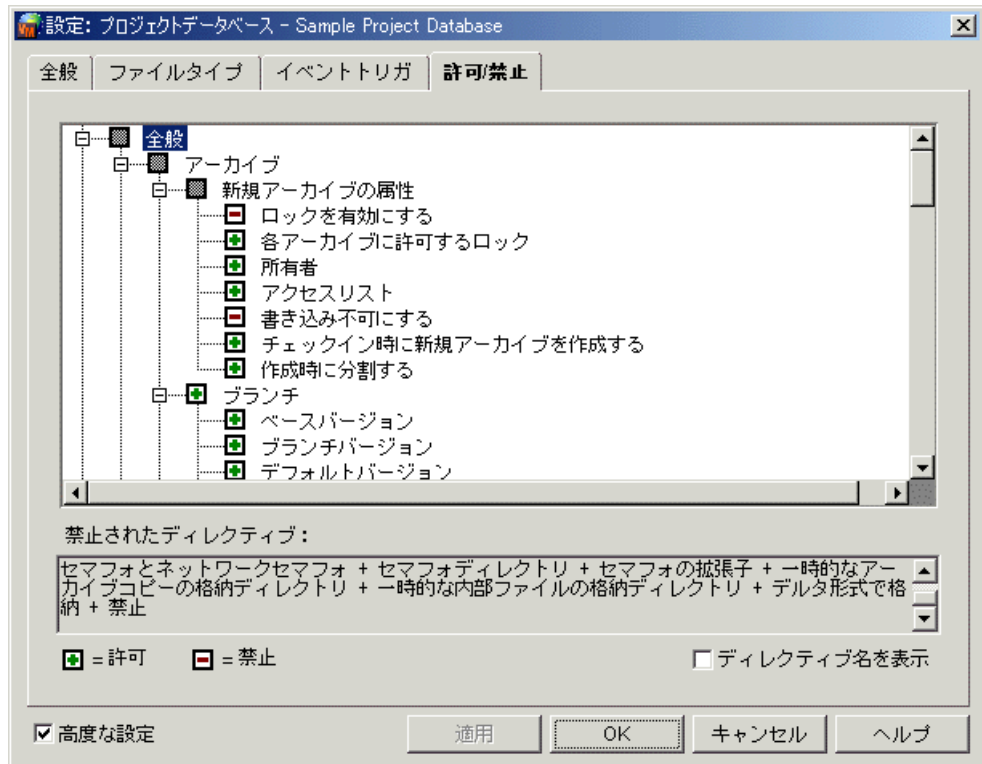
マスターコンフィグレーションファイル中でオプションを変更許可または変更禁止に設定することにより、ユーザが自分のコンフィグレーションファイル中でそのオプションを変更できるかどうかが決まります。オプションを変更禁止に設定すると、ユーザはそのオプションを変更できなくなります。オプションの変更禁止はマスターコンフィグレーションファイル内でのみ設定できます。[許可 / 禁止] タブはプロジェクトデータベースを設定するときのみ使用できます。

ユーザ ID を取得するために Version Manager が使用するソースリスト、アクセスコントロールデータベース、セマフォディレクトリ、ロック設定、および監査を行うときは、ジャーナルオプションなどの設定で、すべてのプロジェクトについて同一の設定を行うことをお勧めします。

デスクトップクライアント

デスクトップクライアントでこれらのオプションを設定するには

- 1 オプションを変更禁止に設定するマスターコンフィグレーションファイルに関連付けられた、プロジェクトデータベースを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示されます。
- 3 〈高度な設定〉 チェックボックスがまだ選択されていない場合は選択します。
- 4 [許可 / 禁止] タブを選択します。



- 5 このタブには変更の許可と禁止を設定できるコンフィグレーションオプションの、チェックボックスを含むツリーが表示されます。マイナス記号 (-) が表示されたオプションは変更禁止、プラス記号 (+) が表示されたオプションは変更許可に設定されています。

オプションのステータスを変更するには、オプションの隣のチェックボックスを選択します。親オプションの選択により、関連するオプション全体の変更の許可と禁止が設定されます。たとえば、「ブランチ」の隣のチェックボックスを選択してマイナス記号 (-) を表示させると、ブランチを変更禁止に設定できます。

- 6 各オプションのディレクティブの名称を表示するには、〈ディレクティブ名を表示〉チェックボックスを選択します。これが選択されていると、オプションの右側にディレクティブの名称がかっこ内に表示されます。
- 7 オプションの定義を終えたら【OK】をクリックするか、【適用】をクリックして設定を保存し、ほかのオプションの定義を続けます。

コマンドラインインターフェイス

コンフィグレーションオプションを変更禁止に設定するためのディレクティブは Disallow です。マスターコンフィグレーションファイル中のディレクティブをすべて変更禁止に設定するには、Disallow をディレクティブと (存在する場合には) ディレクティブの No- 形、たとえば MultiLock と NoMultiLock の両方に対して使用する必要があります。ディレクティブのみを変

更禁止に設定した場合は、ほかのコンフィグレーションファイルからディレクティブの No- 形を指定できます。たとえば MultiLock の使用を禁じるには、次のように指定します。

DISALLOW MULTILOCK NOMULTILOCK



注 デスクトップクライアントでは、ディレクティブを変更許可または変更禁止に設定すると、ディレクティブとその否定形の両方が設定されます。

Version Manager へのマスターコンフィグレーションファイルの埋め込み

Version Manager にコンフィグレーションファイルを埋め込むことにより、すべてのユーザが同じ設定を使用するようになり、ほかのマスターコンフィグレーションファイルは使用できなくなります。Version Manager に埋め込まれたマスターコンフィグレーションファイルは、そのファイルが埋め込まれた Version Manager を使用するすべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。

デスクトップクライアントを使用する場合

Version Manager にはマスターコンフィグレーションファイルを 1 つだけ埋め込むことができ、このコンフィグレーションファイルは、すべてのプロジェクトデータベースに対する Version Manager の動作を制御します。マスターコンフィグレーションファイルを Version Manager に埋め込むと、各プロジェクトデータベースに関連付けたコンフィグレーションファイルは、プロジェクトコンフィグレーションファイルとなります。プロジェクトコンフィグレーションファイルは、マスターコンフィグレーションファイルではなくなるため、このコンフィグレーションファイルでの Disallow オプションは無視されます。

マスターコンフィグレーションファイルを埋め込むには

- 1 Version Manager に埋め込むマスターコンフィグレーションファイルに関連付けられた、プロジェクトデータベースを選択します。

- 2 [管理] - [設定の保護] を選択します。[設定の保護] ダイアログボックスが表示されます。

- 3 <コンフィグレーションファイル> チェックボックスを選択します。このチェックボックスの隣にあるフィールドには、埋め込もうとするコンフィグレーションファイルの場所と名称が表示されます。このフィールドは編集できません。



注 マスターコンフィグレーションファイルの名称を埋め込んだ後、このファイル (VMWFVC.DLL または vmufvc.a) が Version Manager の実行ファイルの場所以外にある場合は、この場所に移動します。また、この場所に対してユーザが書き込み権限を持っていないことを確認します。Version Manager への別のマスターコンフィグレーションファイルの埋め込みや、マスターコンフィグレーションファイルの変更を、ユーザに許可しないようにします。

- 4 【OK】 をクリックします。

コマンドラインインターフェイスを使用する場合

コマンドラインインターフェイスでは、Version Manager にマスターコンフィグレーションファイルとして作成したコンフィグレーションファイルを使用させるために、そのコンフィグレーションファイルの名称を Version Manager に埋め込む必要があります。

ファイル名を埋め込むには次のように VCONFIG コマンドを使用します。

```
vconfig -cconfig_filename vm_filename
```

ここで、

config_filename は、埋め込むコンフィグレーションファイルの名称です。



注 vconfig は、bin ディレクトリのサブディレクトリ、admin にあります。

vm_filename は次のいずれかになります。

- Windows : VMWFVC.DLL

- UNIX : vmufvc.a

マスターコンフィグレーションファイルの名称を埋め込んだ後、このファイル (VMWFVC.DLL または vmufvc.a) が Version Manager の実行ファイルの場所以外にある場合は、この場所に移動します。また、この場所に対してユーザが書き込み権限を持っていないことを確認します。Version Manager への別のマスターコンフィグレーションファイルの埋め込みや、マスターコンフィグレーションファイルの変更を、ユーザに許可しないようにします。

VCONFIG コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

TrackerLink と SourceBridge のセットアップ

SourceBridge は Serena のバージョン管理と案件管理ソリューションを統合します。これにより、ファイルをソース管理からチェックインおよびチェックアウトするとき TeamTrack および Tracker で作成した Issue にバージョン管理ファイルに関連付けることができます。

SourceBridge のセットアップは、どの Version Manager クライアントおよび Issue 追跡システムを使用するかによって異なります (以下の表を参照してください)。



注 TeamTrack SourceBridge を実行するには特定の TeamTrack ユーザ権限が必要です。TeamTrack SourceBridge のマニュアルを参照してください。

VM クライアント	TeamTrack	Tracker
デスクトップクライアント	<p>Version Manager クライアントに SourceBridge コンポーネントをインストールする必要があります。このコンポーネントは、TeamTrack Web クライアントを使用してダウンロードすることができます。</p> <p>そのためには、Web クライアントのツールバーの【バージョン情報】ボタンをクリックします。【バージョン情報】ダイアログボックスから、【製品情報】タブを選択します。〈SourceBridge をインストール〉リンクをクリックします。詳しい手順は、『SourceBridge ユーザガイド』を参照してください。</p> <p>いったん SourceBridge コンポーネントをインストールしたら TeamTrack サーバとログイン情報を指定します。</p> <p>そのためには、[ツール] - [SourceBridge] を選択します (これを最初に行うとき [SourceBridge の設定] ダイアログボックスが表示されます。その後 [SourceBridge] ダイアログボックスで【設定】ボタンをクリックします)。詳細な手順は、『SourceBridge ユーザガイド』を参照してください。</p>	<p>TrackerLink をインストールして設定する必要があります。詳細については、Tracker のマニュアルを参照してください。</p>
Web クライアント	<p>Version Manager 管理者は、案件管理と連携するよう各 Web サーブレットを設定する必要があります。詳細については、Version Manager Application Configuration ユーティリティの【サーブレット】タブの【ヘルプ】ボタンをクリックするか、『Version Manager インストールガイド』を参照してください。</p>	

VM クライアント	TeamTrack	Tracker
SCC IDE	<p>Version Manager クライアントに SourceBridge コンポーネントをインストールする必要があります。このコンポーネントは、TeamTrack Web クライアントを使用してダウンロードすることができます。</p> <p>そのためには、Web クライアントのツールバーの【バージョン情報】ボタンをクリックします。【バージョン情報】ダイアログボックスから、【製品情報】タブを選択します。〈SourceBridge をインストール〉リンクをクリックします。詳しい手順は、『SourceBridge ユーザガイド』を参照してください。</p> <p>いったん SourceBridge コンポーネントをインストールすると、SourceBridge が Source Code Control (SCC) プロバイダに設定されます。デフォルトでは、SourceBridge は、TeamTrack を案件管理プロバイダとして使用し、Version Manager をソース管理プロバイダとして使用します。</p> <p>SourceBridge が使用するソース管理プロバイダは [SourceBridge] ダイアログボックスの [システム設定] タブで指定することができます (ただし、Serena SCC の【テスト】ボタンなど SCC/IDE 環境から起動した場合のみ)。SCC プロバイダは、ChangeMan SCC Admin ユーティリティから変更することができます (Windows の [スタート] メニューから [すべてのプログラム] - [ChangeMan Version Manager] - [Version Manager IDE Client] - [Version Manager SCC Admin] を選択します)。詳細については、『Serena ChangeMan IDE クライアント操作ガイド』を参照してください。</p>	<p>TrackerLink をインストールし、ソースコード管理 (SCC) プロバイダとして TrackerLink を使用するよう IDE を設定する必要があります。詳細については、Tracker のマニュアルを参照してください。</p>
リッチ IDE 統合	<p>Eclipse および .NET とのリッチ IDE 統合により、Version Manager と TeamTrack の機能が IDE に統合されます。詳細については、『Serena ChangeMan IDE クライアント操作ガイド』を参照してください。</p>	<p>不可。リッチ統合は、TrackerLink には対応していません。</p>

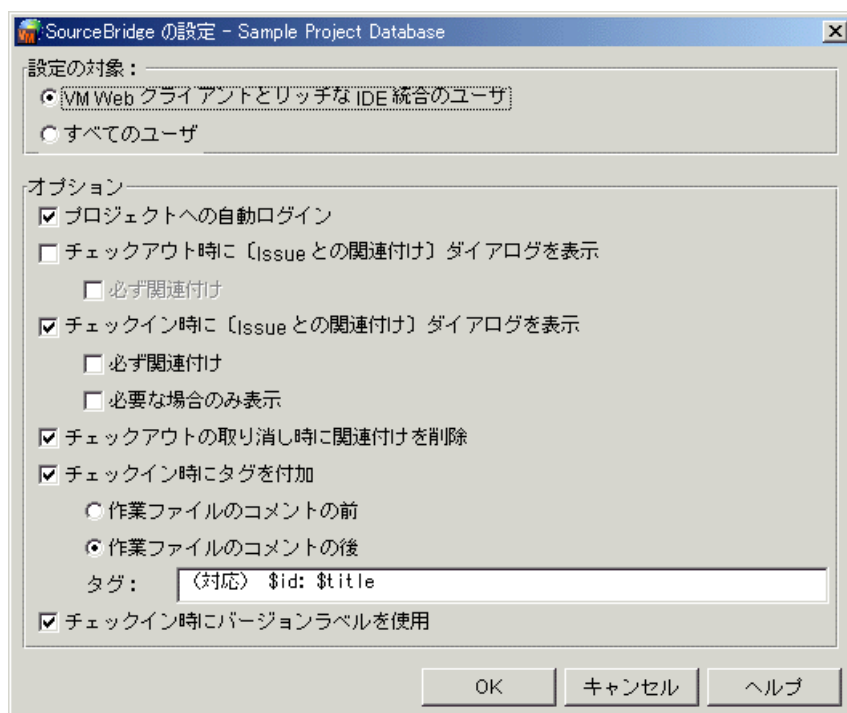
Source Bridge ユーザオプションの設定

SourceBridge オプションは Version Manager で設定され、Superuser 権限を持つ管理者が Version Manager のプロジェクトデータベースに設定します。次の設定はオプションです。選択した設定は、作業ファイルに Issue を関連付ける際に、ユーザに実行させるプロセスの数に影響します。



注 [TrackerLink の設定] ダイアログボックスは、[SourceBridge の設定] ダイアログボックスに変わりました。これは、Tracker TrackerLink および TeamTrack SourceBridge の両方の Serena 案件管理統合に適用されます。

[管理] - [SourceBridge の設定] から、[SourceBridge の設定] ダイアログボックスにアクセスできます。



SourceBridge を Version Manager デスクトップクライアントと Web クライアントのユーザ用に設定するには

SourceBridge の設定はプロジェクトデータベースに存在します。案件管理と使用するプロジェクトデータベースごとに SourceBridge を設定する必要があります。これらの設定はすべての Version Manager ユーザに適用されます。また、Version Manager の Web クライアントおよびリッチ IDE 統合ユーザのみに設定することもできます。リッチ IDE 統合に影響するのは、〈チェックイン時にバージョンラベルを使用〉、〈関連付け時に作業ファイルのコメントにタグを付加〉、および〈必ず関連付け〉 (チェックイン時) の 3 つのフィールドだけです。

- Web クライアントおよびリッチ IDE 統合ユーザにのみ SourceBridge の設定を行うには、〈VM Web クライアントとリッチな IDE 統合のユーザ〉オプションを選択します。
- すべての Version Manager ユーザに SourceBridge の設定を行うには、〈すべてのユーザ〉を選択します。



注 自分のシステムで独自の設定を行っているユーザがいた場合、〈すべてのユーザ〉を選択すると、既存の設定は [SourceBridge の設定] ダイアログボックスで設定した内容で上書きされます。

ユーザの自動ログインを設定するには (TrackerLink のみ)

〈プロジェクトへの自動ログイン〉を選択すると、ユーザは自動的に Tracker にログインします。このオプションが選択されていないと、ユーザが作業ファイルに関連付けを行うたびに、Tracker へのログイン画面が表示されます。

チェックアウトまたはロックの際に SourceBridge を使用するには

- 〈チェックアウト時に [Issue との関連付け] ダイアログを表示〉チェックボックスを選択すると、作業ファイルからのチェックアウトまたはロックの際に、[Issue との関連付け] ダイアログボックスが自動的に表示されます。
- 〈チェックアウト時に [Issue との関連付け] ダイアログを表示〉チェックボックスの下の〈必ず関連付け〉を選択すると、チェックアウトまたはロックするファイルに Issue を関連付けることができます。ユーザがファイルに 1 つ以上の Issue を関連付けるまで、バージョン管理ファイルやリビジョンはロックまたはチェックアウトされません。

作業ファイルのチェックインまたは追加時に SourceBridge を使用するには

- 〈チェックイン時に [Issue との関連付け] ダイアログを表示〉を選択すると、作業ファイルのチェックインまたは追加の際に、[Issue との関連付け] ダイアログボックスが自動的に表示されます。
- 〈チェックイン時に [Issue との関連付け] ダイアログを表示〉チェックボックスの下の〈必ず関連付け〉を選択すると、チェックインまたは追加するファイルに Issue を関連付けることができます。チェックボックスが選択されている場合、[作業ファイルの追加] または [チェックイン] ダイアログボックスで【OK】をクリックすると、自動的に [Issue との関連付け] ダイアログボックスが自動的に表示されます。ユーザがファイルに 1 つ以上の Issue を関連付けるまで作業ファイルは追加またはチェックインされません。
- (SCC IDE のみ) 関連付けダイアログを、1 つ以上のファイルがすでに Issue と関連付けされていない場合にのみ表示するよう限定することができます。そのためには、〈必要な場合のみ表示〉チェックボックスを選択します。

ファイルのロック解除時に関連付けを解除するには

〈チェックアウトの取り消し時に関連付けを削除〉チェックボックスを選択すると、作業ファイルのロックを解除する時に、すべての割り当てられた関連付けを削除します。作業ファイルがチェックインされるまで、関連付けは完全ではありません。この解除オプションは、作業途中の作業ファイルの関連付けを解除するために役立ちます。

Issue との関連付けを作業ファイルのコメントに登録するには

Issue と作業ファイルの関連付けを作業ファイルのコメントフィールドに記録するには、タグを使用します。ファイルを追加する時に、ファイルを Issue と関連付けることができます。アーカイブの作成時に Version Manager が使用するデフォルトの文字列「最初のリビジョン」は、Issue 情報の前または後に付加されます。〈関連付け時に作業ファイルのコメントにタグを付加〉に書式を入力して、コメントの書式および含まれる Issue 情報を選択できます。

- 文字列をコメントの前に挿入するには、〈作業ファイルのコメントの前〉を選択します。
- 文字列をコメントの後に挿入するには、〈作業ファイルのコメントの後〉を選択します。

案件管理を使用してバージョンラベルを付けるには

〈チェックイン時にバージョンラベルを使用〉が選択されていると、各々の Issue 番号を含む、リビジョンに関連付けられたバージョンラベルが作成されます。たとえば、ユーザが foo.b をチェックインし、リビジョン 1.4 を作成します。ユーザはこのファイルに、DEF762 と DEF833 の 2 つの Issue を関連付けます。Version Manager がリビジョンを作成する時に、2 つのバージョンラベルが生成され、リビジョン 1.4 に対応付けられます。

バージョンラベルの書式は Version Manager 内では設定できないことに注意してください。案件管理管理者により設定する必要があります。

TrackerLink および SourceBridge の完全なヘルプを見るには、関連するダイアログボックスの【ヘルプ】ボタンをクリックしてください。

TeamTrack と TrackerLink の併用

一部のプロジェクトデータベースには Tracker を維持しながら、ほかには TeamTrack を使用したい場合があります。Version Manager には、両方の案件管理統合を使用するユーザに配慮していますが、詳細は使用する Version Manager クライアントによって異なります。クライアント固有の詳細は、次のセクションを参照してください。

デスクトップクライアント

次回 Version Manager デスクトップクライアントの起動するときどの案件管理統合を使用するかを設定することができます。現在開かれているクライアントセッションには影響しません。

- 1 Serena Issue Management Integration ユーティリティを起動します (Windows の [スタート] メニューから、[すべてのプログラム] - [Serena] - [ChangeMan Version Manager] - [Issue Management Integration] 選択します)。
- 2 [TeamTrack SourceBridge] または [Tracker TrackerLink] を選択します。
- 3 [OK] または [Version Manager を起動する] ボタンをクリックします。

Web クライアント

各プロジェクトデータベースのサブレットによって、使用する案件管理統合が決定されます。異なる案件管理統合を使用するプロジェクトデータベースを切り替えるために、クライアントシステムで変更するものではありません。

SCC IDE

SourceBridge または TrackerLink を SCC ベースの IDE と使用するには、どちらか 1 つを SCC プロバイダに設定します。

- 1 Serena SCC Admin ユーティリティを起動します (Windows の [スタート] メニューから [すべてのプログラム] - [Serena] - [ChangeMan Version Manager] - [Version Manager IDE Client] - [Version Manager SCC Admin] を選択します)。
- 2 SCC プロバイダの一覧から [Merant TrackerLink] または [Serena SourceBridge] を選択します。
- 3 [OK] をクリックします。
- 4 IDE を再起動します。

Eclipse および .NET とのリッチ IDE 統合

Eclipse および Visual Studio .NET とのリッチ統合は TeamTrack をベースにしています。TrackerLink をリッチ統合と使用することはできません。



注 Eclipse および .NET との SCC 統合には TrackerLink または TeamTrack を使用することができます。

詳細情報

案件管理のインストール、設定、および使用に関する詳細については Tracker と TeamTrack のマニュアルを参照してください。

Serena ChangeMan Mover の設定

Serena
ChangeMan Mover
について

Serena[®] ChangeMan[®] Mover[™] アプリケーションは、コードとアプリケーション開発およびサーバの配布と公開までの溝を橋渡しするものです。Serena Mover を使用すると、ファイルとフォルダの取得や導入をネットワーク全体で自動化することができます。Mover では、指定した基準に基づいて、Version Manager または Dimensions のプロジェクトからファイルが抽出され、任意のサーバの組み合わせに対してファイルが導入されます。

Mover を使用して導入プロセスを管理すると、開発サーバ上のその他すべてのファイルやフォルダの転送も自動化し、ネットワーク上で任意のコマンド（ビルドスクリプトやサーバコマンドなど）をトリガすることもできます。このように、Mover をいったん設定した後は、コードのコンパイルと配布をすべて自動的に処理することができます。

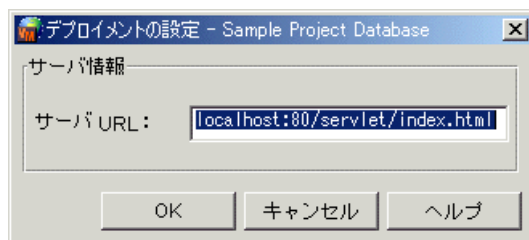
Mover サーバへの接続の設定

Serena ChangeMan Mover サーバに対する接続設定を定義し保存するには、以下の手順に従います。

- 特記事項
- 接続設定はプロジェクトデータベースごとに保存されます。
 - プロジェクトデータベースに関する接続が定義されると、そのデータベースにログインできるすべてのユーザは Mover にログインできるようになります。
 - Mover サーバへの接続を設定するには、Superuser 権限が必要です。

Mover の接続設定を定義するには

- 1 Version Manager で、Mover URL を定義するプロジェクトデータベースを選択します。
- 2 [管理] - [Mover のセットアップ] を選択します。[デプロイメントの設定] ダイアログボックスが表示されます。



- 3 <サーバ URL> フィールドに Mover URL を次のように入力します。

`http://server:port/servlet`

次にその例を示します。

`http://Deployment:80/servlet`

- 4 【OK】 をクリックします。

Version Manager が Mover サーバと接続できない場合、エラーが表示されます。その他の場合、デスクトップクライアントに戻り、Mover にログインすることができます。

Version Manager と Mover の併用

Mover ログインページには、Version Manager、Dimensions、または Web ブラウザから直接アクセスできます。



注 Mover にログインするために特別な Version Manager 権限または Dimensions 権限は必要ありません。

Mover にログインするには

- 1 次のいずれかの作業を行います。

ログイン先	必要な操作
ブラウザ	次の URL に移動します。 <code>http://server:port/servlet</code> たとえば、Mover が Deployment という名のサーバのポート 80 (デフォルト) にインストールされている場合の移動先は次のようになります。 <code>http://Deployment:80/servlet</code>
Version Manager デスクトップクライアント	[管理] - [Mover] を選択します。
Version ManagerWeb クライアント	[アクション] - [Mover] を選択します。

- 2 Mover のログインページで、自分のユーザ名とパスワードを入力します。



注 管理者が事前に Mover にユーザのユーザアカウントを設定している必要があります。

初めてログインする場合は、〈ユーザ名〉フィールドと〈パスワード〉フィールドの両方に「admin」と入力します。

- 3 【ログイン】 ボタンをクリックします。
- 4 【ログインオプション】 ダイアログボックス内で次の作業を行います。

作業	必要な操作
Mover のユーザアクセスの設定、またはプロジェクトの作成	〈管理〉を選択
プロジェクトの作成	〈新規プロジェクト〉を選択
プロジェクトへのログイン	〈プロジェクトを開く〉を選択し、ログインするプロジェクトを選択 重要! 管理者が事前にプロジェクトを作成している必要があります。

- 5 【OK】 をクリックします。

Mover の設定と Version Manager との使用に関する詳細は、Serena 『ChangeMan Mover インストールとセットアップガイド』および『Serena ChangeMan Mover ユーザガイド』およびを参照してください。

トラブルシューティング

このセクションでは、コンフィグレーションファイルの使用時に発生する可能性のある状況について説明します。

コンフィグレーションファイルがほかのユーザによってロックされている

プロジェクトデータベースを開こうとしたときに、そのプロジェクトデータベースがほかのユーザによってロックされているというエラーメッセージが表示され、実際にはそのようなことはない場合には、次の作業を実行します。

- 1 コンフィグレーションファイルと同じディレクトリにある、そのコンフィグレーションファイルの `.lck` ファイルを削除します。
- 2 コンフィグレーションファイルにセマフォファイルがあれば、それを削除します。セマフォファイルが存在するかどうかは、コンフィグレーションファイルの設定によります。コンフィグレーションファイルによりセマフォファイルの作成が指示されている場合は、そのコンフィグレーションファイルに対するセマフォファイルが存在します。セマフォファイルは、コンフィグレーションファイルで指定された場所にあります。セマフォファイルに使用される拡張子も、コンフィグレーションファイル内で定義されています。デスクトップクライアントでのデフォルトの拡張子は `+sem` で、たとえばコンフィグレーションファイルの名称が `cb006nf.cfg` の場合、セマフォファイルの名称は `cb006nf.cfg-sem` になります。

第 4 章

Version Manager File Server の設定と使用

Version Manager File Server について	130
Version Manager File Server のインストール	134
Version Manager File Server の管理について	134
File Server の起動と停止	134
Version Manager File Server の管理ユーティリティの起動	136
管理者ユーザの管理	136
プロジェクトデータベースとリビジョンライブラリのパスの管理	137
File Server の設定	142
サーバステータスの表示	146
サーバログの表示	148
Version Manager File Server へのプロジェクトデータベースの追加	149
リビジョンライブラリの作成	158
アーカイブのエクスポート、インポート、移動、名前変更、および修正	163
クロスプラットフォーム環境における File Server の使用	165
セキュリティに関する特記事項	166
File Server とともに使用するためのクライアントの設定	168

Version Manager File Server について

Version Manager File Server は、Version Manager ユーザに代わって、アーカイブ、メタデータアーカイブ、リビジョン、その他のファイルにアクセスするサーバです。

これは、Version Manager 8 で追加されたオプションの新機能です。

Version Manager File Server を使用する理由

Version Manager File Server を使用すると、Version Manager のセキュリティとパフォーマンスを向上させることができます。

Windows におけるセキュリティの向上

以前は、Version Manager のユーザは、ファイルシステム上のアーカイブやメタデータファイルに対する書き込み / 削除アクセス権が必要でした。この方法では、ユーザが Windows エクスプローラまたは OS のコマンドプロンプトを使用して、故意または誤ってアーカイブやメタデータファイルを削除する危険がありました。

Version Manager File Server を使用すると、アーカイブやメタデータファイルに対する書き込み / 削除アクセス権はサーバ自体にしか必要とならないため、このような脆弱性を解消することができます。



注 Version Manager の Web クライアントはこれまでも、セキュリティで保護されたアーカイブによる利点をもたらしてきましたが、それには CLI、PCLI、およびデスクトップクライアントに対するアクセスを制限できることが必要でした。UNIX 上のすべてのクライアントは、これまでもセキュリティで保護することが可能でした。

パフォーマンスの向上

Version Manager File Server を使用すると、大半のソース制御操作を高速化することができます。最終的なパフォーマンス向上の度合いは、以下に示すとおり、作業および作業慣行の特徴のほか、コンフィグレーションをどれだけ詳細にモデル化するかによって決まります。

- File Server を実行しているシステム上のすべてのアーカイブファイルとメタデータファイルを見つけます。マップされたドライブ経由でのファイルアクセスはしません。これにより、以下が可能になり、時間の節約とネットワークトラフィックの削減を図ることができます。
 - ファイルを一度に 1 ブロックずつ読み取るのではなく、ネットワーク全体でストリームできます。
 - 一部の副次的な操作を File Server で実行し、ファイルをネットワークで転送しなくても済むようにできます。
- アーカイブを別々のリビジョンファイルとメタデータファイルに分割します。これにより、以下が実現されます。
 - メタデータアーカイブはリビジョンを含むアーカイブよりもサイズが大幅に小さいため、メタデータの処理が高速化されます。



注 メタデータには、バージョンラベル、プロモーショングループ、チェックアウト日付、変更記述などが含まれます。

- リビジョンは個々のフラットファイルとして格納できるため、リビジョンの処理が高速化されます。
- リビジョンをデルタストアの外部でキャッシュして検索を高速化できるため、デルタストア内のリビジョンへのアクセスが高速化されます。

Version Manager File Server の互換性

Version Manager File Server の導入は、絶対に必要な作業ではありません。導入は以下のように行うことができます。

- 一部のプロジェクトデータベースにのみ導入し、残りのデータベースには影響を及ぼさない
- ユーザが File Server 経由 (Version Manager 8 以降を使用) および直接的 (Version Manager 5.3 以降を使用) に同じプロジェクトとアーカイブにアクセスするのを可能にする



注 リビジョンライブラリ (分割アーカイブ) には、直接的にも Version Manager 8 より前のリリースを使用してもアクセスすることはできません。リビジョンライブラリにアクセスするには、File Server を使用する必要があります。

- リビジョンライブラリとメタデータアーカイブを再結合 (分割解除) し、Version Manager 5.3 以降で直接アクセスできるもと (非圧縮) のアーカイブフォーマットに戻す。

次の表は、Version Manager File Server のファイルサーバ機能とリビジョンライブラリ機能に対する、Version Manager の各リリースとアーカイブフォーマットの互換性を示したものです。

クライアントリリース	アーカイブフォーマット	ファイルサーバ	リビジョンライブラリ	互換性に関する説明
8 +	8 + (分割) *	○	○	セキュリティとパフォーマンスに関するすべての機能を使用できます。
8 +	5.3 - 8 +	○	不可	すべてのセキュリティ機能と一部のパフォーマンス機能を使用できます。
6.5 - 7.5	5.3 - 8 +	○	不可	これらのクライアントは、マップされたドライブを使用して File Server 上の非分割アーカイブに直接アクセスできます。 ただし、File Server 自体が実際に使用されるわけではないので、パフォーマンスの向上はありません。 Web クライアントを除き、これらのクライアントには、アーカイブファイルに対する書き込み / 削除アクセス権が必要です。これらのアクセス権を許可すると、File Server に加えられた主なセキュリティの強化が損なわれます。
5.3 - 7.5	8 + (分割) *	不可	不可	これらのクライアントは、File Server とともに使用できないため、分割アーカイブを読み取ることはできません。 ただし、アーカイブの分割を解除してもとの非圧縮形式に戻すことができます。

* 注 : Version Manager 8 以降は、Version Manager 5.3 ~ 7.5 と同じアーカイブフォーマットを使用できます。ただし、リビジョンライブラリを作成するには、アーカイブを別々のリビジョンファイルとメタデータファイルに分割する必要があります。これらの分割アーカイブには、Version Manager 5.3 ~ 7.5 クライアントに対する下位互換性はありません。リビジョンライブラリとメタデータアーカイブを再結合 (分割解除) してもと (非圧縮) のアーカイブフォーマットに戻すことができます。

Version Manager File Server の仕組み

Version Manager File Server には、以下の 2 つの主な役割があります。

- **ファイルサーバ** : Version Manager ユーザに代わってアーカイブファイルとメタデータファイルにアクセスします。

これにより、アーカイブファイルに対するファイルシステムの書き込み / 削除アクセス権がユーザにとって不要となるため、セキュリティを強化することができます。

- **リビジョンライブラリ** : リビジョンデータをメタデータと分離して格納します (分割アーカイブ)。

これにより、メタデータの操作でリビジョンデータファイルを開く必要がなくなるため、パフォーマンスを向上させることができます。また、一部の操作はサーバ上で直接行うことができるため、ネットワークトラフィックと処理にかかる時間をとも削減することができます。

リビジョンはリビジョンライブラリに次の形式で格納されます。

- **デルタファイル (delta.d)** : チップリビジョン全体を格納し、その他すべてのリビジョンをチップリビジョンからの変更として記述します。(リバースデルタとも呼ばれます。)

デフォルトで、Version Manager はテキストファイルのみをデルタとして格納します。詳細については、[112 ページ](#)の『[ファイルタイプに基づいて設定されるオプション](#)』を参照してください。

- **非圧縮形式のフラットファイル (RevisionNumberu)** : 各リビジョンを圧縮されていない個別のファイルとして格納します (例 : 1.4u)。

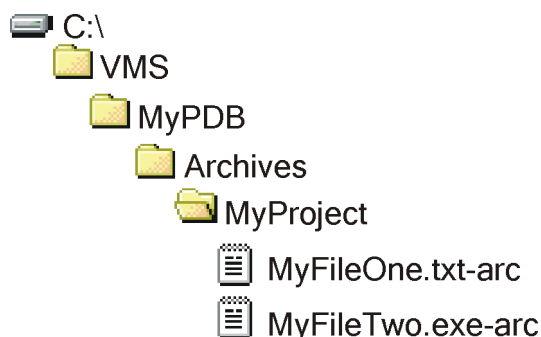


注 delta-todo.d という名前のファイルが作成される場合があります。このファイルは、トラフィックの多いサーバでリビジョンをチェックインした場合に作成されることがあります。このファイルの役割は、現在の形式がフラットファイルであり、デルタファイルに追加する必要がある新しいリビジョンの存在を File Server に伝えることです。サーバによってリビジョンがデルタファイルに追加されると、delta-todo.d ファイルは自動的に削除されます。

リビジョンライブラリに対するすべてのアクセスは File Server によって処理され、それによってすべての機能が正常に機能するため、エンドユーザはサーバ側のこれらの詳細を気にかける必要はありません。ただし、リビジョンライブラリ機能は、Version Manager の 8 より前のリリースまたは圧縮されていない (非分割) アーカイブとは互換性がありません。

ファイルサーバ機能のみ

Version Manager File Server のファイルサーバ機能を導入するが、リビジョンライブラリ機能は導入しない場合、メタデータとリビジョンデータは、ファイルごとに単一の (非圧縮) アーカイブにまとめて格納されます。Version Manager アーカイブはこれまでこのように動作してきましたが、File Server がアーカイブへのアクセスを制御するようになると、次のように格納が行われます。



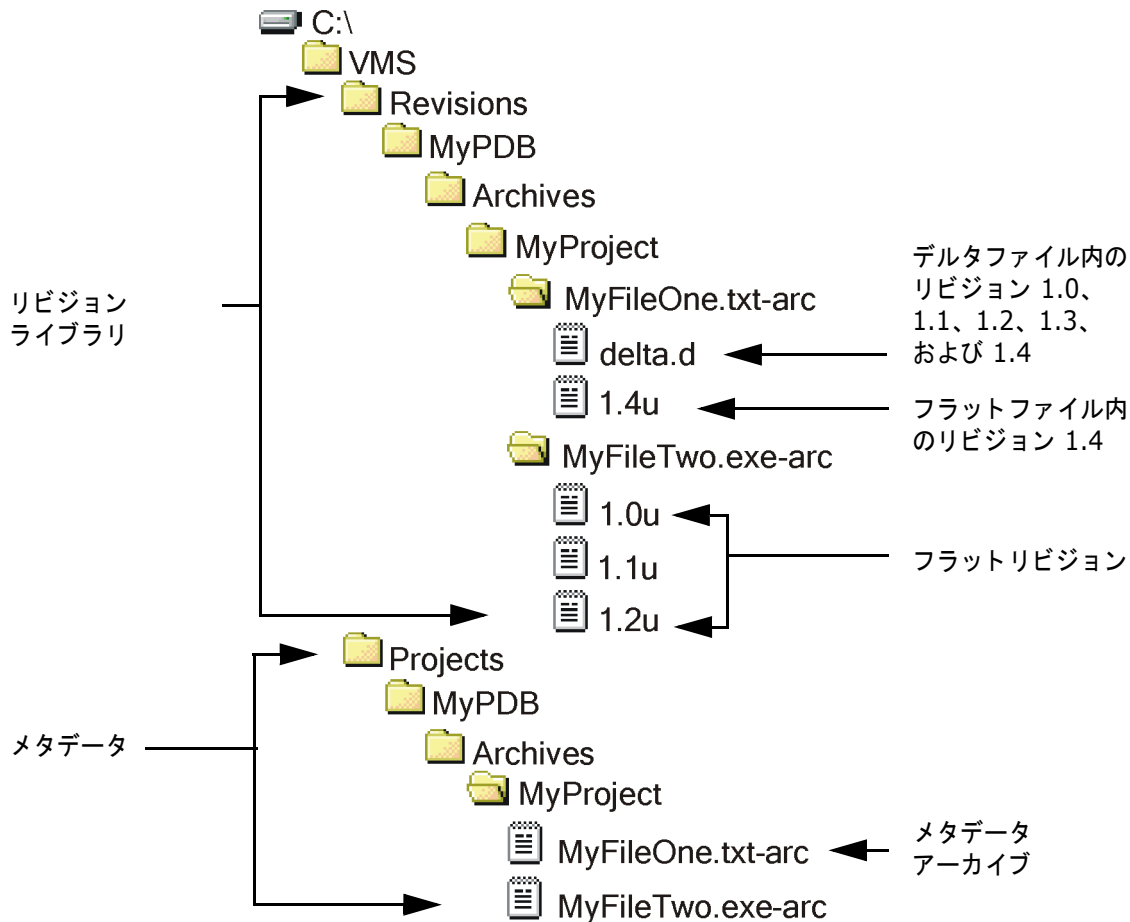
ここで、VMS ディレクトリは、Version Manager File Server にマップされています。VMS ディレクトリ以下のすべてのプロジェクトデータベースは、File Server によって制御されます。

ファイルサーバ機能とリビジョンライブラリ機能

Version Manager File Server のリビジョンライブラリ機能とファイルサーバ機能を導入すると、アーカイブは別々のメタデータコンポーネントとリビジョンコンポーネントに分割され、File Server に格納されます。

- メタデータはメタデータアーカイブに格納されます。
- 各リビジョンは、フラットファイルまたはデルタファイルのエントリとしてリビジョンライブラリに格納されます。

この場合、次の例のようになります。



以下は上記の例の説明です。

- リビジョンパスは C:\¥VMS¥Revisions にマップされています。これは、リビジョンライブラリのルートです。リビジョンライブラリのディレクトリ構造は、メタデータストアの構造とアーカイブファイルの名前に基づいています。この構造を手動で変更してはいけません。
- プロジェクトデータベースのパスは C:\¥VMS¥Projects にマップされています。これは、メタデータストアのルートです。このディレクトリ構造は、リビジョンライブラリと関連しており、手動で変更してはいけません。

- プロジェクトデータベースの名前は MyPDB です。この例でほかのプロジェクトデータベースがあったとすると、MyPDB ディレクトリで始まる構造に似たディレクトリ構造によって表されません。
- プロジェクトの名前は MyProject です。



重要！ メタデータファイルまたはリビジョンデータファイルを手動で移動すると、それらの関係が壊れます。163 ページの『アーカイブのエクスポート、インポート、移動、名前変更、および修正』を参照してください。

Version Manager File Server のインストール

Version Manager File Server は、Version Manager のインストール時に選択できるオプション機能の 1 つです。インストールの方法については、『Serena Version Manager インストールガイド』を参照してください。



重要！ Eclipse は、UNC パスでは正しく動作しません。UNC パスの代わりに、Version Manager インストールのマッピングされたドライブおよび Eclipse ベース IDE と使用する任意の Version Manager ファイルサーバパスを使用してください。

Version Manager File Server の管理について

Version Manager File Server には、Version Manager File Server の管理ユーティリティが含まれています。ブラウザ対応のこのユーティリティを使用すると、以下を行うことができます。

- Version Manager File Server の管理ユーティリティを使用することを許可されたユーザのリストを管理する。136 ページの『管理者ユーザの管理』を参照してください。
- サーバにプロジェクトデータベースをマップする。137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- サーバにリビジョンライブラリをマップする。137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- サーバを設定する。142 ページの『File Server の設定』を参照してください。
- サーバ上で実行される操作のステータスを表示する。146 ページの『サーバステータスの表示』を参照してください。
- サーバエラーのログを表示する。148 ページの『サーバログの表示』を参照してください。

File Server の起動と停止

Version Manager File Server は、Version Manager Application Server 上で実行されるサブプレットです。このサブプレットは、Version Manager Application Server をインストールすると、自動的にインストールされて設定されます。Version Manager File Server を起動または停止するには、Version Manager Application Server を起動または停止します。

File Server はいったん起動されると、手動で停止するか、インストールされているマシンを停止するまでは、継続して実行されます。

Windows における File Server の起動または停止



重要！ この操作を行うと、Version Manager Application Server が起動または停止され、Version Manager File Server、Version Manager Web Server、および WebDAV Server がインストールされている場合は、それらも起動または停止されます。

File Server を Windows で起動するには

- 1 File Server がインストールされているシステムで [スタート] - [すべてのプログラム] - [Serena] - [ChangeMan Version Manager] - [Version Manager Application Server] を選択します。Version Manager Application Server Admin が表示されます。
- 2 [サーバ] タブで [スタート] ボタンをクリックします。



注 【NT サービスに登録】 ボタンをクリックすると、Version Manager Application Server を NT サービスとしてインストールすることもできます。このサービスには、Serena VM Web Application Server という名前が付けられます。

File Server を Windows で停止するには

- 1 File Server がインストールされているシステムで [スタート] - [Serena] - [ChangeMan Version Manager] - [Version Manager Application Server] を選択します。Version Manager Application Server Admin が表示されます。
- 2 [サーバ] タブで [ストップ] ボタンをクリックします。

UNIX における File Server の起動または停止

File Server はいったん起動されると、手動で停止するか、インストールされているマシンを停止するまでは、継続して実行されます。

特記事項

- Version Manager Application Server の起動と停止は、プロジェクトデータベースを所有するユーザとして行います。ルートとして実行しないでください。
- これにより、Version Manager Application Server が起動または停止され、Version Manager File Server、Version Manager Web Server、および WebDAV Server がインストールされている場合は、それらも起動または停止されます。

File Server を UNIX で起動するには

- 1 次のディレクトリに移動します。
`/VM_Install_Dir/vm/common/bin`
- 2 次のコマンドを入力します。
`./pvcsstart.sh`

File Server を UNIX で停止するには

- 1 次のディレクトリに移動します。
`/VM_Install_Dir/vm/common/bin`

- 2 次のコマンドを入力します。

```
./pvcsstop.sh
```

Version Manager File Server の管理ユーティリティの起動

- 特記事項
- Version Manager File Server の管理ユーティリティを起動するには、Version Manager File Server が実行されていることが必要です。134 ページの『File Server の起動と停止』を参照してください。
 - 管理をリモートで行うように File Server を設定することもできます。デフォルトでは、Version Manager File Server の管理ユーティリティはサーバ上でしか実行できません。142 ページの『File Server の設定』を参照してください。

Version Manager File Server の管理ユーティリティを起動するには

- 1 次の URL をブラウザに入力します。

```
http://ServerName:Port/serenafs/Admin
```

各パラメータの意味は次のとおりです。

- *ServerName* は、File Server がインストールされているコンピュータの名前または IP アドレスです。
- *Port* は、File Server に割り当てられたポート番号です。デフォルトで、Version Manager File Server はポート 8080 を使用します。

- 2 ENTER キーを押します。ログインページが表示されます。



ヒント ログインページはブラウザのブックマークに登録してください。

- 3 有効な管理者 ID とパスワードを入力します。



注 初めてログインする場合は、デフォルトのユーザ ID とパスワードを使用します。デフォルトはともに **Admin** です。

- 4 【ログイン】ボタンをクリックします。

管理者ユーザの管理

Version Manager File Server の管理ユーティリティにログインするには、通常の Version Manager ユーザ ID では不十分です。File Server 管理ユーティリティの〔管理者〕ウィンドウで定義されている ID とパスワードを使用する必要があります。

ユーザを管理するには

- 1 Version Manager File Server の管理ユーティリティを起動します。




注 初めてログインする場合は、デフォルトのユーザ ID とパスワードを使用します。デフォルトはともに **Admin** です。

- 2 左側のウィンドウで〈管理者〉を選択します。〔管理者〕ウィンドウが表示されます。
- 3 以下のうち必要な作業を行います。


ユーザの追加

新しいユーザを追加するには

- a 追加 () ボタンをクリックします。〔管理者ユーザの追加〕ダイアログボックスが表示されます。
- b 〈管理者名〉フィールドに新しいユーザ ID を入力します。このユーザ ID は、すでに存在するものであってはならず、改行されない空白文字 () を使用してはいけません。ユーザ ID では、大文字と小文字が区別されます。
- c 〈パスワード〉フィールドにパスワードを入力します。パスワードでは大文字と小文字が区別されます。5 文字以上で指定する必要があります。
- d パスワードが正しく入力されたことを確認するため、〈パスワードの確認〉フィールドにパスワードを再入力します。
- e 【OK】をクリックします。


パスワードの変更

パスワードを変更するには

- a ユーザ ID を選択します。
- b 編集 () ボタンをクリックします。〔管理者ユーザの編集〕ダイアログボックスが表示されます。
- c 〈新しいパスワード〉フィールドに新しいパスワードを入力します。
- d パスワードが正しく入力されたことを確認するため、〈新しいパスワードの確認〉フィールドにパスワードを再入力します。
- e 【OK】をクリックします。

ユーザの削除

ユーザ ID を削除するには

- a ユーザ ID を選択します。
- b 削除 () ボタンをクリックします。確認のためのダイアログボックスが表示されます。



注 現在のログイン時に使用したユーザ ID を削除することはできませんが、自分以外のユーザは現在ログインしているかどうかにかかわらず削除することができます。

- c 【OK】をクリックします。

プロジェクトデータベースとリビジョンライブラリのパスの管理

File Server でアクセスするプロジェクトデータベースとリビジョンライブラリのパスを指定する必要があります。

- 特記事項
- パスを指定する前に、リビジョンパスとプロジェクト DB パスの間の関係を必ず理解してください。132 ページの『Version Manager File Server の仕組み』を参照してください。
 - ユーザーの数が多く、アーカイブのサイズが大きい場合、またはアーカイブの数が多く場合などは、複数の File Server 間で作業負荷を分散させると効果的である場合があります。142 ページの『複数の File Server への作業の割り当て』を参照してください。



重要！ Eclipse は、UNC パスでは正しく動作しません。UNC パスの代わりに、Version Manager インストールのマッピングされたドライブおよび Eclipse ベース IDE と使用する任意の Version Manager ファイルサーバパスを使用してください。

パスを管理するには

- 1 Version Manager File Server の管理ユーティリティを起動します。
- 2 左側のウィンドウで〈パスマップ〉を選択します。[パスマップ] ウィンドウが表示されます。



パスマップ

有効	クライアント名	プロジェクトデータベースパス	リビジョンパス	PDB の公開	アクセスデータベース	ユーザ ID の 大文字と小文字を区別する	LDAP	PDB 作成用の パスワード
<input checked="" type="checkbox"/>	s\WMPDBs	CAVMPDBs	C\WMrVb	<input checked="" type="checkbox"/>	c:\ACDB\access.db	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	y\Alpha\PDB\VM\Alpha	C\Alpha\PDB	C\Alpha\RvLb	<input checked="" type="checkbox"/>	--	--	--	--

- 3 以下のうち必要な作業を行います。
 - パスを追加または編集するには、139 ページの『パスマップの追加または編集』を参照してください。
 - パスマップのセキュリティオプションを設定するには、141 ページの『パスマップセキュリティオプションの設定』を参照してください。
 - パスマップを削除するには、142 ページの『パスマップの削除』を参照してください。

パスマップの追加または編集

パスを追加または編集するには

- 1 新しいパスを追加するには、追加（）ボタンをクリックします。既存のパスを編集するには、パスを選択して編集（）ボタンをクリックします。[パスマップの追加] または [パスマップの編集] のダイアログボックスが表示されます。

全般フィールド

セキュリティフィールド

有効:

クライアント名: s:\VMPDBs

プロジェクトデータベースパス: C:\VMPDBs

リビジョンパス: C:\VMRvLb

PDBの公開:

アクセスコントロールデータベース: c:\ACDB\access.db

ユーザ ID の大文字と小文字を区別する:

LDAP 認証:

PDB 作成用のパスワード: **

パスワードの確認: **

OK キャンセル ヘルプ



注 ここでは全般フィールドについて説明します。セキュリティフィールドについては、141ページの『パスマップセキュリティオプションの設定』を参照してください。

- 2 〈許可〉リストで「有効」を選択し、サーバパスを有効にして、クライアントがそれを使用できるようにします。
- 3 〈クライアント名〉フィールドに仮想パスを入力します。これは、クライアントで表示されるパスになります。このパスは一意的なものである必要があります。複数の仮想パスを区切るには、セミコロン (;) を使用します (UNIX クライアントと Windows クライアントにそれぞれ 1 つずつ指定する場合など)。

特記事項

- 既存のプロジェクトデータベースのパスマップを作成する最も簡単な方法は、プロジェクトデータベースの既存のパスと一致するように〈クライアント名〉を設定することです。たとえば、マップされたネットワークドライブ上のディレクトリ (S:\¥VMPDBS) に既存のプロジェクトデータベースがある場合、〈クライアント名〉は「S:\¥VMPDBS」などに設定します。

- クライアント名は、実際のパスに対応している必要はありませんが（上記のように既存のプロジェクトデータベースをマップしている場合を除く）、クライアントに対する正当な OS パスまたは UNC パスと同じ形式にする必要があります。UNC パスは、二つの円記号（¥¥）で始める必要があります。
- クライアント名は、関連するプロジェクトデータベースパスが互いに同じ方法で関連している場合にのみ、別のクライアント名の親または子になることができます。たとえば、次のようなパスを使用できます。

クライアント名	プロジェクトパス
¥¥client¥parent	C:¥project¥one
¥¥client¥parent¥child	C:¥project¥one¥child

- パスに UNIX と Windows の両方のクライアントからアクセスする必要がある場合は、各 OS の正しい形式でクライアント名をセミコロンで区切って入力するか、Windows スタイルのパスを使用して Windows と UNIX の両システムからパスにアクセスすることができます。

クロスプラットフォーム環境での作業の詳細については、[165 ページの『クロスプラットフォーム環境における File Server の使用』](#)を参照してください。

- 4 単一のプロジェクトデータベースのパス、または複数のプロジェクトデータベースを含むディレクトリ構造のルートへのパスを指定するには、〈プロジェクト DB パス〉フィールドにパスを入力します。このパスにセミコロン (;) を使用してはいけません。
- 5 〈リビジョンパス〉フィールドで、リビジョンライブラリの作成場所とするディレクトリへのパスを入力します。このパスにセミコロン (;) を使用してはいけません。



重要！ アーカイブへの完全なパスがオペレーティングシステムのパス名の長さ制限を超える可能性があるため、長い〈プロジェクト DB パス〉または〈リビジョンパス〉値を指定しないでください。

- 6 プロジェクトデータベースをプロジェクトデータベースパス内で検索し、検索結果のリストにクライアントがアクセスできるようにするには、〈PDB の公開〉フィールドで「有効」を選択します。これにより、ユーザはクライアントで表示されるリストからプロジェクトデータベースを選択できるようになります。この機能を有効にしないと、パスを知っていて手動で入力できるユーザだけがプロジェクトデータベースにアクセスできません。



注 公開プロジェクトデータベースリストを作成する際に検索するディレクトリ構造の範囲を制限する方法については、[142 ページの『File Server の設定』](#)を参照してください。

- 7 【OK】をクリックします。



重要！ ここに記載される手順だけを行っても、Version Manager File Server に新規または既存のプロジェクトデータベースに追加されるわけではありません。[149 ページの『Version Manager File Server へのプロジェクトデータベースの追加』](#)を参照してください。



パスマップセキュリティオプションの設定



注 ここでは、[パスマップの追加] ダイアログボックスと [パスマップの編集] ダイアログボックスのセキュリティフィールドについて説明します。全般フィールドについては、[139 ページの『パスマップの追加または編集』](#)を参照してください。

パスマップセキュリティの設定はオプションですが、設定することをお勧めします。

パスマップセキュリティオプションを設定するには

- 1 新しいパスを追加するには、追加（）ボタンをクリックします。既存のパスを編集するには、パスを選択して編集（）ボタンをクリックします。[パスマップの追加] または [パスマップの編集] のダイアログボックスが表示されます。
- 2 **アクセスコントロールデータベース**：パスマップに関連付けるアクセスコントロールデータベースを指定します。いったんアクセスコントロールデータベースを指定すると、パスマップのすべてのアクセスがそれを使用して確認されます。有効なユーザ ID がないとパスマップのプロジェクトデータベースを作成することができません。存在しないアクセスコントロールデータベースを指定すると、だれもパスマップにアクセスすることができなくなります。



注

- このアクセスコントロールデータベースは、パスマップのアクセスのみ管理します。プロジェクトデータベース内の既存アクセスコントロールデータベースを置き換えたり、クライアントに特権あるいはユーザグループ情報を渡したりはしません。
- アーカイブ名とパスワードは、パスマップアクセスコントロールデータベースとプロジェクトデータベースに対して有効なアクセスコントロールデータベースの両方と一致する必要があります。

- 3 **ユーザ ID で大文字と小文字を区別**：アクセスコントロールデータベースが大文字と小文字を区別したユーザ ID を要求するかを指定します。デフォルトでは、このオプションは有効になっています。
- 4 **LDAP 認証**：パスマップの LDAP 認証が有効かどうかを指定します。デフォルトでは無効です。Version Manager File Server は、次のファイルから設定情報を探します。

```
VM_Install\vm\common\bin\OS\pvcsldap.ini
```



注 パスマップの LDAP 認証を有効にするには、サーバの LDAP 認証を設定する必要があります。[95 ページの『LDAP コンフィグレーションオプション』](#)を参照してください。

- 5 **PDB 作成のパスワード**：プロジェクトデータベース作成操作にパスワードを要求する場合、ここで指定します。
- 6 **パスワードの確認**：パスワードを正しく入力したことを確認するため再入力します。

詳細情報


Version Manager セキュリティの設定に関する詳細は、次の表を参照してください。

参照内容	参照先
ファイルサーバのセキュリティオプションの設定	142 ページの『File Server の設定』

参照内容	参照先
ファイルサーバのセキュリティに関する特記事項	166 ページの『セキュリティに関する特記事項』
Version Manager セキュリティの設定	201 ページの『セキュリティの使用』

パスマップの削除

パスマップを削除するには

- 1 削除するパスマップを選択します。
- 2 削除 () ボタンをクリックします。確認のためのダイアログボックスが表示されます。
- 3 【OK】をクリックします。

複数の File Server への作業の割り当て

ユーザの数が多く、アーカイブのサイズが大きい場合、またはアーカイブの数が多く、または、複数の File Server 間で作業負荷を分散させると効果的である場合があります。

単一または複数のプロジェクトデータベースを独自の File Server 上に配置します。通常の手順を使用して、プロジェクトを設定し、それを File Server にマップします。ただし、どのプロジェクトデータベースを特定の File Server に配置するかを決める際は、サーバ間で負荷の均衡が最適となるように、容量や使用率のパターンを考慮する必要があります。

File Server の設定

File Server の各機能の有効化、無効化、設定を行うと、ハードウェア、OS、使用率などに基づいてパフォーマンスとセキュリティを最適化することができます。

File Server を設定するには

- 1 Version Manager File Server の管理ユーティリティを起動します。[136 ページの『Version Manager File Server の管理ユーティリティの起動』](#)を参照してください。

- 2 左側のウィンドウで〈オプション〉を選択します。[サーバオプション] ウィンドウが表示されます。

- 3 以下のうち必要な変更を行います。

一般的なオプション

- **デバイス固有の一時ディレクトリ**：ファイルの転送時に使用するテンポラリディレクトリのリスト。このリストの最初のエンタリは、デフォルトのテンポラリディレクトリです。それ以降のエンタリは、特定のドライブへのアクセスや書き込みの際に使用するディレクトリを指定します。パフォーマンスを最高にするには、Version Manager File Server によって公開されたプロジェクトデータベースまたはアーカイブを含むドライブ / ファイルシステムごとにテンポラリディレクトリを指定してください。

複数のエンタリはセミコロン (;) を使用して、C:¥temp;D:¥temp;T:¥VMtemp のように区切ります。

- **クライアントパスの大文字と小文字を区別する**：大文字と小文字を区別するクライアントパス名を指定できます。デフォルトでは、このオプションは無効になっています。UNIX 環境では、このオプションを有効にすることによってパフォーマンスを上げることができます。以下のすべての条件が満たされない限りこのオプションを有効にしないでください。
 - ファイルサーバが UNIX システム上にある。
 - すべてのクライアントが UNIX システム上にある。

- Version Manager File Server の管理ユーティリティに定義されているすべてのクライアント名が、UNIX 形式のパスになっている（Windows 形式または UNC 形式クライアント名がない）。
- **リビジョンライブラリの場所の確認**：アーカイブの分割後に移動または名前の変更が行われているかどうかを確認するために、メタデータアーカイブをチェックするようクライアントに指示します。デフォルトでは無効になっています。
- **ログレベル**：どのイベントをログに書き込むかを指定します。以下のオプションがあります。
 - **警告とエラーのみ**（デフォルト）
 - **詳細**：警告、エラー、および情報メッセージを含みます。情報メッセージは、SUCCESS が戻されなかったすべての API 呼び出しに関して生成されます。これには、呼び出し元が SUCCESS が戻されることを期待していなかった呼び出しも含まれます。たとえば、呼び出し元がディレクトリの作成を確認するためにそのディレクトリに対して PUT コマンドを発行する場合、予想される応答は FsDuplicatePathException です。
- **PDB ルート階層の最大値**：公開されたプロジェクトデータベースパス内でプロジェクトデータベースを検索する際の深度をディレクトリ数で表したものの。これらのプロジェクトデータベースはクライアントに表示され、ユーザが選択できます。指定できるオプションは、**0、1、2**（デフォルト）、**3、4、5、6、7、8、9、10**、および無制限です。
0 は、プロジェクトデータベースパスの最後の要素がプロジェクトデータベースディレクトリである場合にのみ選択してください。

パフォーマンスに関するオプション



注 この機能は、〈PDB の埋め込みを許可〉オプションの影響を受けます。

- **PDB の埋め込みを許可**：デフォルトで、Version Manager は、〈PDB ルート階層の最大値〉オプションで指定された深度に達するか、プロジェクトデータベースが見つかるかのどちらかが起こるまで、公開されたプロジェクトデータベースパス内を検索します。ほかのプロジェクトデータベースの下位にネストされたプロジェクトデータベースを公開するには、この機能を有効にします。こうすることにより、途中でプロジェクトデータベースが見つかったも、〈PDB ルート階層の最大値〉オプションで指定した値に達するまで検索が継続されません。

特記事項

- 大きなディレクトリ構造で検索深度を深く設定すると、パフォーマンスが損なわれる可能性があります。
- プロジェクトデータベースはネストしないことをお勧めします。
- 既存のプロジェクトデータベースの下にプロジェクトデータベースを作成することはできません。
- 既存のプロジェクトデータベースの上にプロジェクトデータベースを作成することはできません。
- **PDB キャッシュの期限**：公開されたプロジェクトデータベースのリストをキャッシュする期間を指定します。この期間を過ぎるとリストは無効になります。この時間的制約に加え、パスマップが変更されたり、〈PDB ルート階層の最大値〉または〈PDB の埋め込みを許可〉のオプションの値が変更された場合にも、キャッシュは無効になります。キャッシュが無効になった場合、公開されたプロジェクトデータベースのリストに対する次のクライアント要求が行われると、更新されたリストがサーバで自動的に生成されます。

オプションは次のとおりです。

- **今すぐ更新**：キャッシュを直ちに更新し、このフィールドの値を「変更時」にリセットします。

- **0 秒** : キャッシュはありません。公開されたプロジェクトデータベースのリストは、クライアントが要求するたびにサーバによって再生成されます。
 - **30 秒** : キャッシュは 30 秒間保持されます。
 - **1 分** : キャッシュは 1 分間保持されます。
 - **5 分** : キャッシュは 5 分間保持されます。
 - **10 分** : キャッシュは 10 分間保持されます。
 - **毎日 HH:MM:** に : キャッシュは毎日 **00:00** と **23:00** の間の指定時にリフレッシュされます。
 - **変更時** : プロジェクトデータベースが作成または削除されるたびにキャッシュは無効になります (デフォルト)。
- **ファイルシステム再試行間隔** : I/O エラーからの回復を試みる際の最大待ち時間 (秒単位)。オプションは、**0 秒**、**1 秒**、**5 秒**、**10 秒**、および **30 秒** (デフォルト) です。
 - **セマフォ再試行間の遅延** : セマフォの取得を試行する際の間隔 (秒単位)。オプションは、**0.1 秒** (デフォルト)、**0.2 秒**、**0.5 秒**、**1 秒**、**5 秒**、および **10 秒** です。
 - **セマフォ再試行回数の最大値** : エラーを戻す前にセマフォの取得を試行する最大回数。オプションは、**1**、**3**、**5**、**10**、**20**、**50** (デフォルト)、**100**、**200**、および無制限です。
 - **フラットファイルの期限** : 未アクセスのリビジョンがデルタストアにも含まれる場合にそれをフラットファイルとして保持する日数。オプションは、**0 日**、**1 日**、**5 日**、**10 日** (デフォルト)、および **30 日** です。
- セキュリティに関するオプション
- **書き込み権限を許可** : リビジョンとメタデータのアーカイブに書き込む操作をユーザが実行することを許可します。このオプションを使用すると、バックアップ操作時に書き込みアクセスを無効にすることができます。デフォルトでは、書き込みアクセスは有効になっています。
 - **認証要求** : Version Manager File Server に対する各要求を認証し、有効な Version Manager クライアントから要求が寄せられたことを確認します。これは、データのセキュリティを確保するために行われます。データをこのように保護しないと、不正なプログラムがデータにアクセスしたり、データを破損したりする可能性があります。このセキュリティ機能では多少のオーバーヘッドが発生しますが、この機能は有効にしたままにすることを強くお勧めします。ほかのセキュリティ措置を完全に信頼できる場合以外は、認証機能は無効にしないでください。
 - **リモートの設定を許可** : ユーザがほかのコンピュータから Version Manager File Server の管理ユーティリティにアクセスできるようにします。デフォルトでは無効になっています。
 - **ユーティリティの使用を許可** : VSPLIT、VTRANSFER、ReadDB、および MakeDB の各ユーティリティの使用を制御します。オプションは次のとおりです。
 - **有効** : すべてのユーティリティを有効にします。
 - **無効** : すべてのユーティリティを無効にします。
 - **読み取り専用** : VSPLIT と MakeDB を無効にし、ReadDB を有効にし、VTRANSFER のエクスポートだけを有効にします。
 - **ローカル専用** : File Server のローカルユーザに対してすべてのユーティリティを有効にします (デフォルト)。
 - **リモートは読み取り専用** : File Server のローカルユーザに対してすべてのユーティリティを有効にし、リモートユーザに対しては VTRANSFER のエクスポートだけを有効にします。

デフォルト - [パスワードの追加] ダイアログボックス

- **LDAP キャッシュの期限** : LDAP ユーザ ID とパスワードをキャッシュする期限を指定します。オプションは **1 分**、**5 分**、**30 分**、および **60 分** です (デフォルト)。
- **アクセスコントロールデータベース** : [パスワードの追加] ダイアログボックスのフィールドにクライアントパスに関連付けるデフォルトアクセスコントロールデータベースが読み込まれます。
- **ユーザ ID の大文字と小文字を区別する** : デフォルトで [パスワードの追加] ダイアログボックスのオプションを有効にします。このオプションにより、アクセスコントロールデータベースが大文字と小文字を区別したユーザ ID を要求するかを指定します。
- **LDAP 認証** : デフォルトで [パスワードの追加] ダイアログボックスのオプションを有効にします。このオプションにより、パスワードの LDAP 認証が有効かどうかを指定します。
- **PDB 作成用のパスワード** : [パスワードの追加] ダイアログボックスのフィールドにプロジェクトデータベース作成操作のデフォルトパスワードが読み込まれます。
- **パスワードの確認** : [パスワードの追加] ダイアログボックスのフィールドに上記〈PDB 作成のパスワード〉フィールドに入力したエントリを確認するデフォルトのパスワードが読み込まれます。
- **統計の期間** : [ステータス] ウィンドウの「操作時間」テーブルで表示するデータをメモリに保持する時間の長さ。オプションは、**10 分**、**1 時間** (デフォルト)、**1 日**、および **2 日** です。

データサイズがステータスログに割り当てられたメモリ量を上回ると、完了したすべての操作のステータスは消去されます。

統計に関するオプション

- 4 デフォルト設定を復元するには、【リセットデフォルト】ボタンをクリックします。
- 5 変更を適用するには、【適用】ボタンをクリックします。

WorkDir ディレクティブと ArchiveWork ディレクティブの設定

File Server にマップされているディレクトリに既存の OS ドライブマッピングを使用してアクセスできる場合を除き、WorkDir ディレクティブと ArchiveWork ディレクティブは、このディレクトリには設定しないでください。File Server 自体を使用してもディレクトリにアクセスすることはできません。これらのディレクティブの設定の詳細については、[110 ページの『テンポラリファイルのオプション』](#)を参照してください。

サーバステータスの表示

Version Manager File Server の管理ユーティリティを使用すると、File Server のステータスに関する 2 種類の情報を表示できます。その 1 つは、一定期間に実行された操作の記録であり、もう 1 つはアクティブなセマフォをファイルパスごとに一覧にしたものです。デスクトップクライアントからは、File Server に対する接続のステータスを表示できます。

Version Manager File Server の管理ユーティリティを使用したサーバステータスの表示

サーバのステータスを表示するには

- 1 Version Manager File Server の管理ユーティリティを起動します。
- 2 左側のウィンドウで〈ステータス〉を選択します。〔サーバの状態〕ウィンドウが表示されます。

The screenshot shows the 'Version Manager File Server' administration page in a Microsoft Internet Explorer browser. The address bar shows 'http://localhost:8080/serenafs/MerantFSAdminServlet'. The page title is 'Version Manager File Server'. On the left, there is a navigation menu with icons for 'ステータス' (Status), 'パスマップ' (Path Map), 'オプション' (Options), 'ログ' (Log), and '管理者' (Admin). The main content area is titled 'サーバのステータス' (Server Status) and includes a refresh interval dropdown set to '実行しない' (Do not execute) and a '更新' (Refresh) button. Below this, there is a table titled '最新 1 時間での操作時間 (秒単位)' (Operation time in the last 1 hour (seconds)).

操作	# 完了	最短	最長	平均時間
Get Configuration	1	0.000	0.000	0.000
Get File	149	0.000	0.130	0.005
Put File	2	0.010	0.030	0.020
Unlock File	1	0.000	0.000	0.000
Append File	7	0.000	0.030	0.007
Get Directory Contents	2	0.000	0.070	0.035
Delete File	1	0.010	0.010	0.010

Below the table, there is a section titled 'セマフォ' (Semaphore) with a sub-section 'ファイルパス' (File Path) showing '有効な詳細なし' (No active details).

3 以下の情報が表示されます。

- 「操作時間」テーブルには、タイトル部分に表示された期間における以下の情報が表示されます。
 - 操作：実行された操作の名前。
 - # **Completed**：特定の操作が実行された回数。
 - 最短：特定の操作が実行されるのにかった時間の最小値。
 - 最長：特定の操作が実行されるのにかった時間の最大値。
 - 平均時間：特定の操作が実行されるのにかった時間の平均。

「操作時間」テーブルの統計の期間を設定する方法については、[142 ページの『File Server の設定』](#)を参照してください。

- 「セマフォ」テーブルには、アクティブなセマフォのファイルパスが表示されます。




4 〔サーバの状態〕ウィンドウを更新する頻度を指定するには、〈更新間隔〉フィールドのオプションのうち 1 つを選択します。選択可能なオプションは、更新しない（デフォルト）、**1 分**、**5 分**、**10 分**、**15 分**、**30 分**です。

5 〔サーバの状態〕ウィンドウを更新するには、【更新】ボタンをクリックします。

- 6 デッドロック状態になったセマフォを解放するには、適切な〈ファイルパス〉エントリの隣にあるラジオボタンを選択し、削除 (X) ボタンをクリックします。すべてのセマフォを解放するには、〈すべてのセマフォ〉エントリの隣にあるラジオボタンを選択し、削除ボタンをクリックします。

デスクトップクライアントを使用したサーバステータスの表示

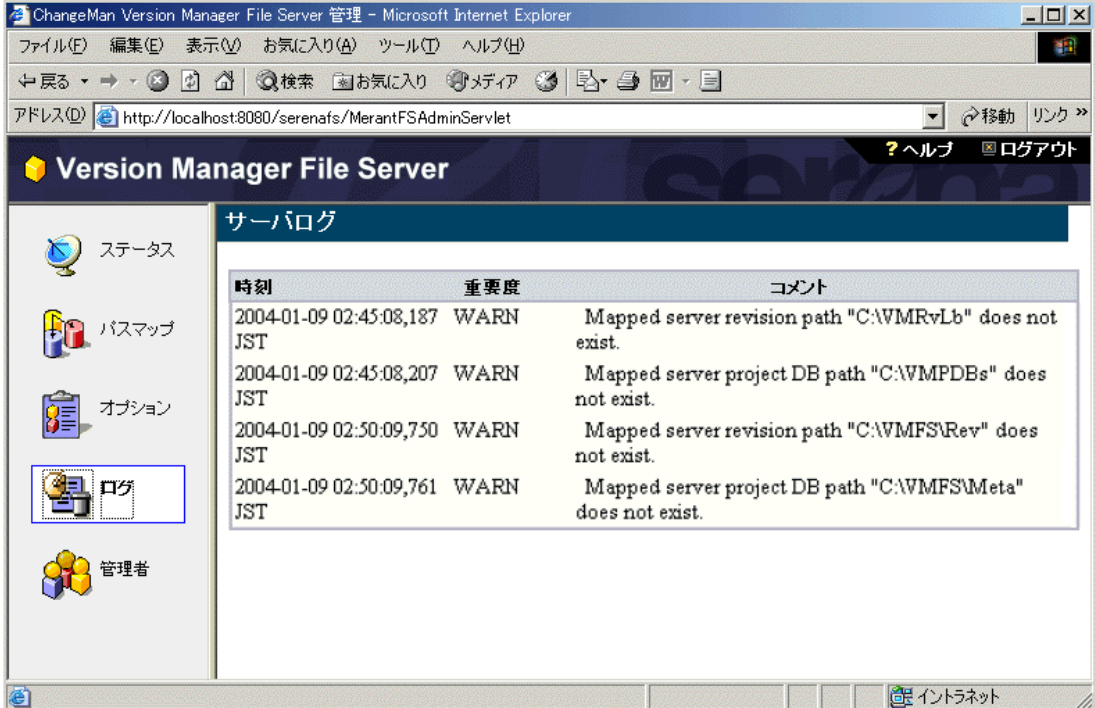
Version Manager デスクトップクライアントでは、サーバステータスは、ステータスバーの左隅に以下のように表示されます (クリックすると、ステータスが更新され、詳細が表示されます)。

-  File Server が設定されていません。
-  1 つまたは複数の設定済み File Server が応答していません。
-  すべての設定済み File Server が正しく動作しています。

サーバステータスは、クライアントがサーバ上のファイルにアクセスしようと試みるたびに更新されます。

サーバログの表示

Version Manager File Server の管理ユーティリティの〔ログ〕ウィンドウでは、エラーや警告などのサーバイベントのログを表示することができます。



The screenshot shows the 'Version Manager File Server' management interface. The 'サーバログ' (Server Log) window is active, displaying a table of log entries. The table has three columns: '時刻' (Time), '重要度' (Severity), and 'コメント' (Comment). The log entries are as follows:

時刻	重要度	コメント
2004-01-09 02:45:08,187 JST	WARN	Mapped server revision path "C:\VMRvLb" does not exist.
2004-01-09 02:45:08,207 JST	WARN	Mapped server project DB path "C:\VMFPDBs" does not exist.
2004-01-09 02:50:09,750 JST	WARN	Mapped server revision path "C:\VMFMS\Rev" does not exist.
2004-01-09 02:50:09,761 JST	WARN	Mapped server project DB path "C:\VMFMS\Meta" does not exist.

各イベントに関して以下の情報が表示されます。

- **時刻**：イベントが検出された日時です。時刻は、サーバのタイムゾーンに基づいています。
- **重要度**：エラー、警告、情報など、イベントの性質を示します。

- **説明**：イベントについて説明し、関連したプログラムなどを記載します。

[ログ] ウィンドウに表示される情報は、次の場所にある pvcsfs.log ファイルに格納されます。

```
InstallLocation¥Serena¥vm¥common¥tomcat¥logs¥
```

ログファイルのサイズが 100KB に達すると、名前が変更され、新しい pvcsfs.log ファイルが作成されます。サーバには最高 10 個のログファイルが保持されますが、[ログ] ウィンドウには現在のログファイルだけが表示されます。古いログファイルを表示するには、テキストエディタを使用します。

Version Manager File Server へのプロジェクトデータベースの追加

プロジェクトデータベースを Version Manager File Server に追加する際にどの方法が最適であるかは、次の質問に対する答えによって決まります。

#	質問	「はい」の場合	「いいえ」の場合
1	既存のプロジェクトデータベースを Version Manager File Server に追加しますか？	質問 2 へ進んでください。	158 ページの『File Server 上で新しいプロジェクトデータベースを作成する方法』を参照してください。
2	プロジェクトデータベースを現在の場所に維持し、データベースを参照するドライブマッピングまたは UNC パスを保持しますか？	149 ページの『既存のプロジェクトデータベースを移動せずに File Server へ追加する方法』を参照してください。	質問 3 へ進んでください。
3	デスクトップクライアントを使用してプロジェクトデータベースを File Server にコピーしますか？	153 ページの『デスクトップクライアントを使用してプロジェクトデータベースを File Server にコピーする方法』を参照してください。	155 ページの『PCLI を使用してプロジェクトデータベースを File Server にコピーする方法』を参照してください。

共有アーカイブ どのアーカイブが共有されているかを判断する方法については、Serena ナレッジベースの記事 14160 を参照してください（記事内容は英語です）。

既存のプロジェクトデータベースを移動せずに File Server へ追加する方法

これは、既存のプロジェクトデータベースを File Server に追加する最も簡単な方法です。既存のプロジェクトデータベース内の情報は、ワークスペース定義や共有アーカイブへの参照も含め、すべて保持されます。

特記事項 Version Manager File Server に切り替える際は、プロジェクトデータベースを新しい場所に移動したり、既存のドライブマッピングを UNC パスに切り替えたりする操作を簡単に行うことができません。File Server への移動時にそのような変更を行う場合は、この手順を使用しないでください。その他のオプションについては、149 ページの『Version Manager File Server へのプロジェクトデータベースの追加』を参照してください。

既存のプロジェクトデータベースを File Server に追加するには

- 1 File Server に追加する予定のプロジェクトデータベースをバックアップします。
- 2 プロジェクトデータベースを含むシステムに Version Manager File Server をインストールします。



注 プロジェクトデータベースのすべてのユーザに対して Version Manager クライアントを Version Manager 8 以降にアップグレードします。

- 3 Version Manager File Server の管理ユーティリティを使用して、プロジェクトデータベースのパスマップを作成します。

〈クライアント名〉をプロジェクトデータベースに対する既存のパスと一致するように設定します。たとえば、マップされたネットワークドライブ (S:¥VMPDBS) に既存のプロジェクトデータベースがある場合は、クライアント名は S:¥VMPDBS などに設定します。[137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』](#)を参照してください。
- 4 Version Manager デスクトップクライアントを File Server に接続します。[168 ページの『File Server とともに使用するためのクライアントの設定』](#)を参照してください。
- 5 (必須ではないが推奨) プロジェクトデータベースに当初アクセスしたときに使用したネットワークマッピングを削除または制限します。

詳細については以下の例を参照してください。

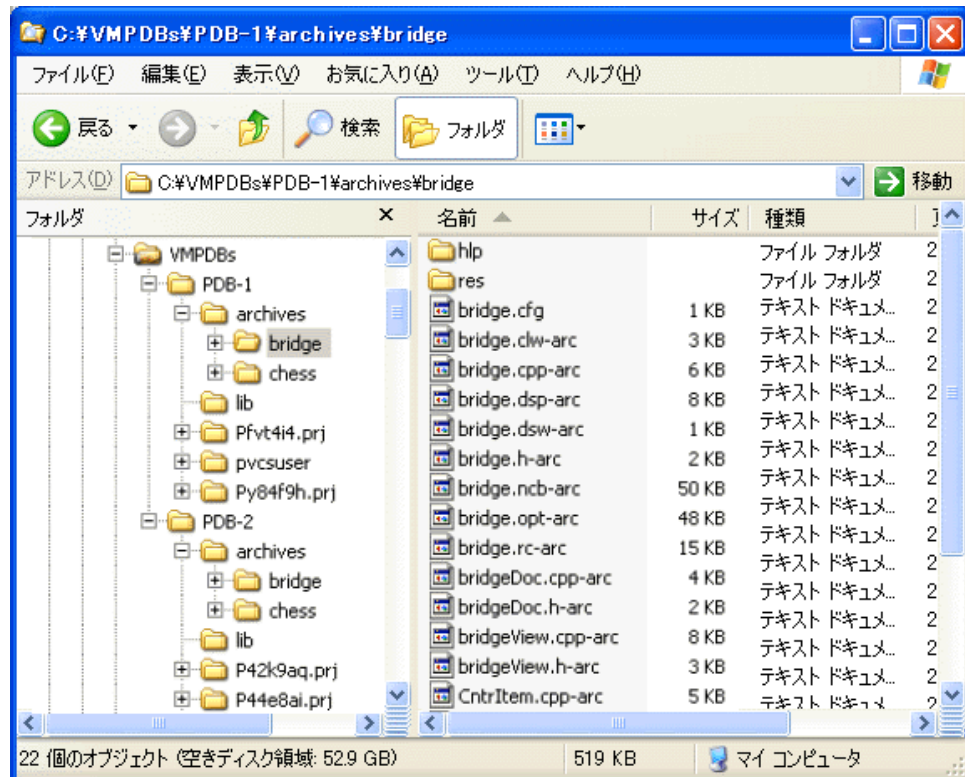
例：既存のプロジェクトデータベースを移動せずに追加する場合

この例では、既存のプロジェクトデータベースである PDB-1 と PDB-2 が、S: というネットワークマッピングを介してアクセスされるシステムの C: ドライブにあると仮定しています。

初期状態

次の図は、プロジェクトデータベースを Version Manager File Server に追加するためにまだ何も行われてない状態の、プロジェクトデータベースファイルの初期位置を示したものです。実際には、

既存のアーカイブを分割してリビジョンライブラリも作成しない限り、この構造は一切変化しません。




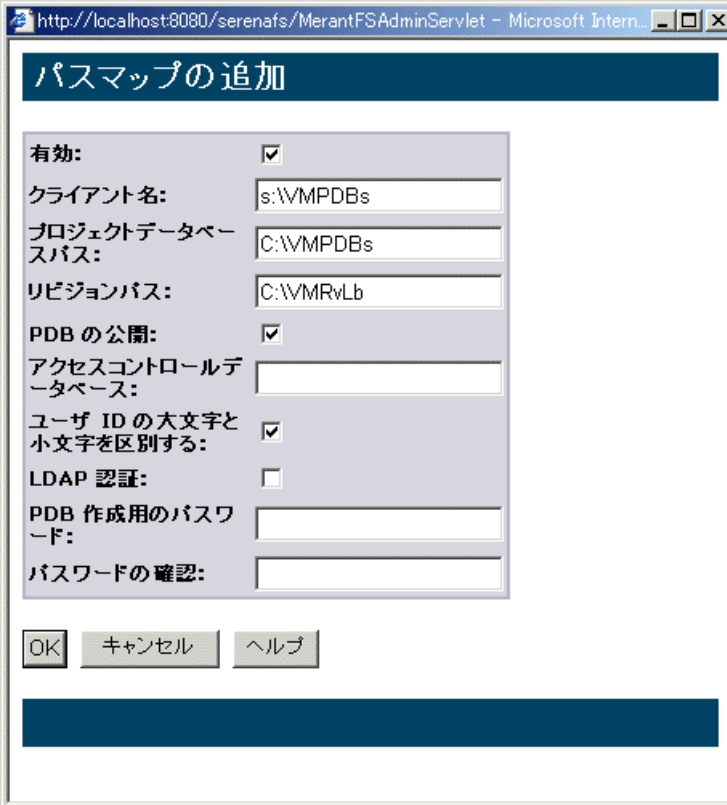
ユーザは現在、次のようにネットワークを介してこれらのプロジェクトデータベースにアクセスしています。

- S:¥VMPDBS¥PDB-1
- S:¥VMPDBS¥PDB-2

ここで、S: は、プロジェクトデータベースをホストするシステム上の C: ドライブの OS ドライブ マッピングです。

- | | |
|-----------------------|--|
| データをバックアップする | 1 File Server に追加する予定のプロジェクトデータベースをバックアップします。 |
| File Server をインストールする | 2 プロジェクトデータベースを含むシステムに Version Manager File Server とデスクトップクライアントをインストールします。Version Manager のインストールについては、『Version Manager インストールガイド』を参照してください。 |
| File Server を起動する | 3 File Server がインストールされているシステムで [スタート] - [すべてのプログラム] - [Serena] - [ChangeMan Version Manager] - [Version Manager Application Server] を選択します。Version Manager Application Server Admin が表示されます。
4 [サーバ] タブで【スタート】ボタンをクリックします。 |
| プロジェクトデータベースパスをマップする | 5 Version Manager File Server の管理ユーティリティを起動し ([Serena ChangeMan Version Manager] - [Serena ChangeMan Version Manager File Server Admin]) し、【パスマップ】ボタンをクリックします。パスマップ表示部が表示されます。 |

- 6 追加（）ボタンをクリックします。[パスマップの追加] ダイアログボックスが表示されます。



- a <許可> リストから「有効」を選択し、パスマップを有効にします。
- b <クライアント名> フィールドに、既存のプロジェクトデータベースの上位にあるディレクトリへのネットワークパスを入力します。この例の場合は S:¥VMPDBS です。



注 パスに UNIX と Windows の両方のクライアントからアクセスする必要がある場合は、各 OS の正しい形式でクライアント名をセミコロンで区切って入力するか、Windows スタイルのパスを使用して Windows と UNIX の両システムからパスにアクセスすることができます。クロスプラットフォーム環境での作業の詳細については、165 ページの『[クロスプラットフォーム環境における File Server の使用](#)』を参照してください。

- c <プロジェクト DB パス> フィールドに、既存のプロジェクトデータベースの上位にあるディレクトリへの OS ネイティブパスを入力します。この例の場合は C:¥VMPDBS です。
- d <リビジョンパス> フィールドに、リビジョンライブラリを作成する場所の下位にあるディレクトリへの OS ネイティブパスを入力します。
- e <PDB の公開> リストから「有効」を選択し、ユーザがクライアントに表示されたリストからプロジェクトデータベースを選択できるようにします。
- f 【OK】をクリックします。

- クライアントを設定する 7 各クライアントワークステーションが File Server に接続するように設定します。168 ページの『[File Server とともに使用するためのクライアントの設定](#)』を参照してください。

プロジェクトデータベースをクライアントから開く

- 8 デスクトップクライアントから、[ファイル] – [プロジェクトデータベースを開く] を選択します。[プロジェクトデータベースを開く] ダイアログボックスが表示されます。



- 9 <検索先> リストから、一番上の項目である「File Servers」を選択します。[パスマップ] ウィンドウでマップした場所の下位にあるすべてのプロジェクトデータベースがダイアログボックスに一覧表示されます。
- 10 プロジェクトデータベースを選択し、【OK】をクリックします。Version Manager クライアントから File Server にアクセスする方法については、172 ページの『Version Manager クライアントから File Server にアクセスする方法』を参照してください。
- 11 ほかのプロジェクトデータベースや作業ファイルへのアクセスで必要とされない限り、プロジェクトデータベースにアクセスするために当初使用したネットワークドライブ（例では S:）はここで削除してかまいません。また、ユーザは今後 Version Manager File Server からしかプロジェクトデータベースにアクセスしなくなるため、OS 権限を使用して、プロジェクトデータベースのディレクトリに対するアクセスを制限することもできます。

この例は、161 ページの『例：アーカイブの分割』の出発点として使用できます。

デスクトップクライアントを使用してプロジェクトデータベースを File Server にコピーする方法

これは、既存のプロジェクトデータベースを別のシステム上の File Server に追加したり、別のパスを使用して追加したりする最も簡単な方法です。



注 これでデスクトップクライアントを使用してプロジェクトデータベースをコピーするとき共有アーカイブとワークスペースを保持することができます。以前は、PCLI のみでこれが可能でした。

前提条件 プロジェクトデータベースを File Server にコピーする前に、以下を行う必要があります。

- 1 File Server をホストするシステムに Version Manager File Server とデスクトップクライアントをインストールします。Version Manager のインストールについては、『Version Manager インストールガイド』を参照してください。



注 プロジェクトデータベースのすべてのユーザに対して Version Manager クライアントを Version Manager 8 以降にアップグレードします。

- 2 Version Manager File Server を起動します。134 ページの『File Server の起動と停止』を参照してください。


- 3 Version Manager File Server の管理ユーティリティを使用して、プロジェクトデータベースのパスマップを作成します。137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- 4 Version Manager デスクトップクライアントを File Server に接続します。168 ページの『File Server とともに使用するためのクライアントの設定』を参照してください。

手順 既存のプロジェクトデータベースを File Server にコピーするには

- 1 コピーする予定のプロジェクトデータベースをバックアップします。
- 2 Version Manager デスクトップクライアントを起動し、[編集] - [コピー] を選択します。[コピープロジェクトデータベース] ウィザードが表示されます。



重要！ クライアントは Version Manager 8 以降である必要があります。

- 3 コピーするプロジェクトデータベースの名前を〈名前〉フィールドに入力します。これは、Version Manager クライアントに表示される名前になります。
- 4 〈場所〉フィールドの隣にある参照ボタンをクリックします。[プロジェクトデータベースの場所の選択] ダイアログボックスが表示されます。
- 5 〈検索先〉リストから「File Servers」を選択します。公開するよう定義しマークされたすべてのクライアント名がダイアログボックスに一覧表示されます。クライアント名の作成については、137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- 6 プロジェクトデータベースの追加先とするクライアント名（パスマップ）を選択します。
- 7 このクライアント名を単一のプロジェクトデータベースのルートとして機能させない限り、プロジェクトデータベースのサブディレクトリを作成する必要があります。そうしない場合、その他のプロジェクトデータベースをこのクライアント名に正しく追加できなくなります。
プロジェクトデータベースに新しいサブディレクトリを作成するには、新規フォルダ () ボタンをクリックし、表示されるダイアログボックスで新しいディレクトリの名前を入力します。
- 8 【OK】をクリックします。[コピープロジェクトデータベース] ダイアログボックスが、〈場所〉と〈アーカイブの場所〉の各フィールドに値が入力された状態で再び表示されます。
- 9 プロジェクトデータベースの〈作業ファイルの場所〉を指定します。



重要！ 作業ファイルの場所は、ネットワークドライブのマッピングまたは UNC パスを介してクライアントからもアクセス可能でない限り、Version Manager File Server にマップされた場所であってはなりません。

- 10 【次へ】をクリックします。〈プロジェクトの作成〉ウィザードの 2 ページ目が表示されます。
- 11 〈プロジェクトの場所にアーカイブをコピー〉オプションを選択します。
- 12 共有アーカイブとの関連を保持するには、〈共有アーカイブの保持〉チェックボックスを選択します。
- 13 〈サブプロジェクトを含む〉チェックボックスを選択します。
- 14 【次へ】をクリックします。〈プロジェクトの作成〉ウィザードの 3 ページ目が表示されます。

- 15 〈既存のコンフィグレーションファイルをコピー〉オプションを選択します。
- 16 〈アクセスコントロールを他の場所にコピー〉オプションを選択します。
- 17 現在のワークスペース定義を保持するには、〈ワークスペースをコピー〉チェックボックスを選択します。
- 18 【完了】ボタンをクリックします。コピー処理の進捗状況に続き結果が表示されます。【詳細】ボタンをクリックし、エラーを調べます。

プロジェクトデータベースが File Server に配置されました。プロジェクトデータベースのリビジョンライブラリを作成するには、[158 ページの『リビジョンライブラリの作成』](#)を参照してください。

PCLI を使用してプロジェクトデータベースを File Server にコピーする方法

これは、既存のプロジェクトデータベースを別のシステム上の File Server に追加したり、別のパスを使用して追加したりする最も完全な方法です。これにより、より複雑な Issue を扱うことができます。



注 これでデスクトップクライアントを使用してプロジェクトデータベースをコピーするとき共有アーカイブとワークスペースを保持することができます。以前これが可能なのは、PCLI だけでした。

[153 ページの『デスクトップクライアントを使用してプロジェクトデータベースを File Server にコピーする方法』](#)を参照してください。

特記事項 この手順はほかのオプションよりも複雑です。以下を行う必要があります。

- PCLI の ExportPDB コマンドを使用して、プロジェクトデータベース情報をテキストファイルにエクスポートする。
- Version Manager File Server における新しいプロジェクトデータベースの場所を反映するように、テキストファイルでパスを編集する。
- プロジェクトデータベースファイルを File Server 上のディレクトリ構造にコピーする。
- Version Manager File Server における新しいプロジェクトデータベースの場所を反映するように、すべてのコンフィグレーション (.CFG) ファイルでパスを編集する。
- PCLI の ImportPDB コマンドを使用して、テキストファイルで編集したパスをプロジェクトデータベースにインポートする。

作業を開始する前に手順全体に目を通すようにしてください。この手順で使用されるコマンドの詳細については、『Version Manager PCLI ユーザガイド & リファレンスガイド』と Serena ナレッジベースの記事 39987 を参照してください。

その他のオプションについては、[149 ページの『Version Manager File Server へのプロジェクトデータベースの追加』](#)を参照してください。

前提条件 プロジェクトデータベースを File Server にコピーする前に、以下を行う必要があります。

- 1 File Server をホストするシステムに Version Manager File Server とデスクトップクライアントをインストールします。Version Manager のインストールについては、『Version Manager インストールガイド』を参照してください。



注 プロジェクトデータベースのすべてのユーザに対して Version Manager クライアントを Version Manager 8 以降にアップグレードします。

- 2 Version Manager File Server を起動します。134 ページの『File Server の起動と停止』を参照してください。
- 3 Version Manager File Server の管理ユーティリティを使用して、プロジェクトデータベースのパスマップを作成します。137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- 4 Version Manager デスクトップクライアントを File Server に接続します。168 ページの『File Server とともに使用するためのクライアントの設定』を参照してください。

手順 **既存のプロジェクトデータベースを File Server にコピーするには**

- 1 コピーする予定のプロジェクトデータベースをバックアップします。
- 2 埋め込まれたコンフィグレーションオプションがどのようにエクスポートとインポートの操作に影響するか考慮してください。続ける前に必要に応じてオプションの埋め込みを解除してください。
- 3 コマンドプロンプトで次のコマンドを入力します。

```
pcli exportpdb -prPathToPDB -idUserID:PW -z -fPathToExportFile
```

各パラメータの意味は次のとおりです。

- *PathToPDB* は、プロジェクトデータベースのルートへの既存パスです。
 - *UserID* は、SuperUser 権限を持つ有効な Version Manager ユーザの ID です。*PW* は関連付けられたパスワードです (ある場合)。これらの情報は、アクセスコントロールデータベースが有効になっている場合にのみ必要となります。
 - *PathToExportFile* は、既存のプロジェクトデータベースからエクスポートされた情報を含むファイルへのパスであり、その名前でもあります。
- 4 エクスポート後のファイルをテキストエディタで開き、Version Manager File Server の管理ユーティリティでマップしたクライアント名に対する File Server の相対パスで既存のパスを置換します。



重要! WorkPath は、ネットワークドライブのマッピングまたは UNC パスを介してクライアントからもアクセス可能でない限り、Version Manager File Server にマップされている場所に設定しないでください。

- 5 ツールまたは OS のコマンドを使用して前のステップでパス参照を変更したすべてのファイルをコピーします。ファイルは、必ずパス参照を変更したクライアント名と一致するファイルサーバ上の場所へコピーしてください。

通常コピーするファイルには次のものがあります。

- すべてのコンフィグレーションファイル (.cfg)。
- すべてのアクセスコントロールデータベースファイル (.db)。

- すべてのアーカイブ (-arc)。



注 上記のファイル拡張子はデフォルトですが、別の拡張子が使用されている場合もあります。

コピーしないファイルとディレクトリ：

- .ser ファイル (pvcsroot.ser、pvcsid.ser など)。
 - .old ファイル (pvcsroot.old、pvcs.old など)。
 - pvcsuser サブディレクトリ
 - lib サブディレクトリ (そこに明示的にファイルを置いていない場合)。
 - P*.prj ワイルドカードと一致するサブディレクトリ。たとえば、Pnoprw.prj など。
- 6 エクスポートファイルで参照されるすべてのコンフィグレーション (.CFG) ファイルを開き、Version Manager File Server の管理ユーティリティでマップしたクライアント名に対して File Server の相対パスで既存のパスを置換します。これらのコンフィグレーションファイルによって参照されるすべてのコンフィグレーションファイルに対してこの手順を繰り返します。



重要! WorkPath は、ネットワークドライブのマッピングまたは UNC パスを介してクライアントからもアクセス可能でない限り、Version Manager File Server にマップされている場所に設定しないでください。

- 7 (省略可能) Version Manager クライアントから過去のジャーナル項目を表示する場合は、.CFG ファイルで行ったようにジャーナルファイルのパスを更新する必要があります。ただし、クライアントで新規項目を表示する場合はこのステップは不要です。その場合にも古い項目はテキストエディタで表示することができます。
- 8 コマンドプロンプトで次のコマンドを入力します。

```
pcli importpdb -prPathToNewPDB -idUserID:PW -fPathToExportFile
```

各パラメータの意味は次のとおりです。

- *PathToNewPDB* は、File Server 上にある新しいプロジェクトデータベースの場所のルートへのパスです。このパスは、クライアント名に対する相対パスを指定する必要があります。
- *UserID* は、SuperUser 権限を持つ有効な Version Manager ユーザの ID です。PW は関連付けられたパスワードです (ある場合)。これらの情報は、アクセスコントロールデータベースが有効になっている場合にのみ必要となります。
- *PathToExportFile* は、既存のプロジェクトデータベースからエクスポートされた情報を含むファイルへのパスであり、その名前でもあります。




注 プラットフォームデータベースの作成にパスワードが必要な場合、-pa パスワードオプションを指定する必要があります。

- 9 必要に応じて上記で埋め込みを解除したコンフィグレーションオプションを再度埋め込みます。

File Server 上で新しいプロジェクトデータベースを作成する方法

File Server 上で新しいプロジェクトデータベースを作成するには

- 1 Version Manager デスクトップクライアントを起動し、[管理] - [プロジェクトデータベースの作成] を選択します。[プロジェクトデータベースの作成] ダイアログボックスが表示されます。
- 2 新しいプロジェクトデータベースの名前を〈名前〉フィールドに入力します。これは、Version Manager クライアントに表示される名前になります。
- 3 〈場所〉フィールドの隣にある参照ボタンをクリックします。[プロジェクトデータベースの場所の選択] ダイアログボックスが表示されます。
- 4 〈検索先〉リストから「File Servers」を選択します。有効になっているすべてのクライアント名がダイアログボックスに一覧表示されます。クライアント名の作成については、[137 ページ](#)の『プロジェクトデータベースとリビジョンライブラリのパスの管理』を参照してください。
- 5 プロジェクトデータベースの追加先とするサーバの場所（パスマップ）を表すクライアント名を選択します。
- 6 このクライアント名を単一のプロジェクトデータベースのルートとして機能させない限り、プロジェクトデータベースのサブディレクトリを作成する必要があります。そうしない場合、その他のプロジェクトデータベースをこのクライアント名に正しく追加できなくなります。

プロジェクトデータベースに新しいサブディレクトリを作成するには、新規フォルダ () ボタンをクリックし、表示されるダイアログボックスで新しいディレクトリの名前を入力します。

- 7 【OK】をクリックします。[プロジェクトデータベースの作成] ダイアログボックスが、〈場所〉と〈アーカイブの場所〉の各フィールドに値が入力された状態で再び表示されます。
- 8 通常プロジェクトデータベースを新規作成する際と同様に、残りのフィールドにも値を指定します。残りのフィールドの詳細については、[24 ページ](#)の『プロジェクトデータベースの作成』を参照してください。

プロジェクトデータベースが File Server に配置されました。プロジェクトデータベースのリビジョンライブラリを作成するには、[158 ページ](#)の『リビジョンライブラリの作成』を参照してください。

リビジョンライブラリの作成

非圧縮アーカイブ これまで、Version Manager では、リビジョンデータとメタデータが同じアーカイブにまとめて格納されていました。これらのアーカイブは、「非圧縮アーカイブ」または「非分割アーカイブ」と呼ばれるようになりました。

分割アーカイブ Version Manager File Server のリビジョンライブラリ機能では、リビジョンデータがメタデータとは分離されて格納されるため、多くの操作が高速化されます。リビジョンはリビジョンライブラリに格納され、メタデータは別のメタデータアーカイブに格納されるため、このように使用されるアーカイブは「分割アーカイブ」と呼ばれます。

分割アーカイブには、Version Manager 8 以降を使用して Version Manager File Server からのみアクセスできます。

RFSSplitOnCreate ディレクティブ	アーカイブを「分割解除」して非圧縮アーカイブフォーマットに復元することができます。復元した非圧縮アーカイブは、Version Manager の旧バージョンと下位互換性があります。 RFSSplitOnCreate コンフィグレーションファイルディレクティブを使用すると、新しいアーカイブは作成時に分割されます。リビジョンライブラリを作成する場合は、プロジェクトデータベースが既存の非圧縮アーカイブから File Server に追加された場合、このディレクティブを設定する必要があります。新規に作成されたプロジェクトデータベースで、このディレクティブが有効になっています。
VSPLIT コマンド	VSPLIT コマンドは、非圧縮アーカイブを別々のリビジョンデータ（リビジョンライブラリ）とメタデータに分割します。

前提条件

リビジョンライブラリを作成する前に以下を行う必要があります。

- Version Manager File Server の管理ユーティリティを使用して、プロジェクトデータベースのリビジョンパスマップを作成する。[137 ページの『プロジェクトデータベースとリビジョンライブラリのパスの管理』](#)を参照してください。
- プロジェクトデータベースを File Server に追加する。[149 ページの『Version Manager File Server へのプロジェクトデータベースの追加』](#)を参照してください。

RFSSplitOnCreate ディレクティブの有効化

RFSSplitOnCreate ディレクティブを使用すると、新しいアーカイブが作成時に分割され、別々のリビジョンデータ（リビジョンライブラリ）とメタデータになります。このディレクティブは、既存のアーカイブには影響を及ぼしませんが、プロジェクトデータベースに追加される新しいアーカイブには適用されます。

新しいアーカイブは、プロジェクトデータベースがリビジョンパスにマップされている場合にのみ分割されます。

RFSSplitOnCreate ディレクティブを有効にするには

- 1 Version Manager デスクトップクライアントを起動し、プロジェクトデータベースを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトデータベースの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 <オプション> リストボックスで、「アーカイブ」の下の「新規アーカイブの属性」を選択します。
- 4 <作成時に分割する> チェックボックスを選択します。
- 5 【OK】 をクリックします。



注 RFSSplitOnCreate ディレクティブをコマンドラインから有効化 / 無効化する方法については、『Version Manager コマンドラインリファレンスガイド』を参照してください。

既存のアーカイブの分割

VSPLIT は、コマンドラインユーティリティであり、既存の非圧縮アーカイブを別々のメタデータストアとリビジョンストアに分割し、Version Manager File Server のリビジョンライブラリ機能で使用できるようにします。VSPLIT は、リビジョンライブラリ内のアーカイブを分割解除してもとの非圧縮アーカイブフォーマットに戻すこともできます。

構文

```
vsplit [Options] Path . . . . .
```

ここで、*Path*は、Version Manager File Server の管理ユーティリティの [パスマップ] ウィンドウで定義したプロジェクトデータベースを表すクライアント名です。

オプション

@ @[*list_file*]

このオプションにより、追加のコマンドラインオプションを記述した *list_file* を読み込みます。Version Manager は、指定したファイルから追加のオプションを読み込んでから、コマンドラインのそれ以外の部分を読み込みます。別のコマンドからの入力を転送する場合は、@ オプションをパラメータを付けずに使用します。

ファイルの拡張子が .GRP である場合は、パスと拡張子を省略できます。Version Manager は、VCSDir ディレクティブで指定されたディレクトリ内でファイルを検索します。

- h コマンドに関するヘルプを表示します。このコマンドは、ほかのオプションを指定した場合でも、-h を処理した後で終了します。
- L アーカイブを一覧表示しますが、分割は行いません。
- r 再帰的な動作（ディレクトリパスに適用されます）。
- u 分割解除。分割アーカイブからリビジョンおよびメタデータを再結合してもとの非圧縮アーカイブフォーマットに戻すことができます。再構成したアーカイブはメタデータの場所にあります。リビジョンライブラリは削除されます。



ヒント RFSSplitOnCreate ディレクティブを無効にしないと、追加する新しいファイルはすべて分割アーカイブに入れられます。

-v このツールのバージョンを表示します。

-xe *file_name*

ステータス、プログラム、およびエラーメッセージを *file_name* に転送します。

-xo *file_name*

標準出力を *file_name* にリダイレクトします。

- 特記事項
- VSPLIT は、File Server を実行するシステムから実行される場合でも、クライアント（パスマップのクライアント名によって定義）によって表示されるアーカイブパスを参照する必要があります。
 - VSPLIT は、既存のアーカイブにのみ影響し、新しいアーカイブの作成には影響を及ぼしません。アーカイブを作成時に分割するには、RFSSplitOnCreate ディレクティブを設定します。

- ほかのプロジェクトデータベースファイルは、VSPLIT によって使用されず、その影響も受けません。
- VSPLIT は、〈ユーティリティの使用を許可〉オプションによって有効になっている場合にのみ実行できます。142 ページの『File Server の設定』を参照してください。
- VSPLIT は、Version Manager のインストール場所の `vm\os\bin\admin` ディレクトリにあります。

VSPLIT 使用のヒント

次の操作を行うようお勧めします。

- **ユーザアクセスを無効にする**：分割するアーカイブをユーザがすでに利用できる場合、実行する前にユーザアクセスを無効にするか制限します。いったん操作を完了して結果を確認したら、再びユーザアクセスを有効にすることができます。VSPLIT をファイルサーバ自体から実行し、アーカイブがサーバのローカルアーカイブである場合、単にサーバをネットワークから切断するだけでユーザアクセスを無効にできます。
- **テスト**：実際の本番環境にできるだけ近いテスト環境を設定します。このテスト環境を使用してプロジェクトデータベースのコピーに対してアップグレードプロセスをテストします。
- **バックアップ**：VSPLIT を実行する前にデータをバックアップします。
- **出力をリダイレクト**：出力を分析して適切に分割されていないアーカイブを確認するため VSPLIT コマンドの出力をファイルにリダイレクトします。出力をファイルヘリダイレクトするには、`-xo` および `-xe` オプションを使用します。これらは、`-xo+e` のように組み合わせることができます。次にその例を示します。
`vsplit -xo+eOutPut.txt -r S:¥VMFS¥PDBS`

この出力の分析に関する詳細については、162 ページの『VSPLIT 出力の分析』を参照してください。

例：アーカイブの分割

この例では、既存のデータベースである PDB-1 と PDB-2 がネットワークマッピングである S: を使用してアクセスされるシステム上にあったと仮定しています。これらのプロジェクトデータベースは、Version Manager File Server にクライアント名のパスマップ S:¥VMPDBS として追加されました。

この例は、150 ページの『例：既存のプロジェクトデータベースを移動せずに追加する場合』の続きに当たります。ここまでの手順とプロジェクトデータベースの初期状態については、前の例を参照してください。

ユーザは現在、Version Manager File Server 上の S:¥VMPDBS パスマッピングを次のように使用して、これらのプロジェクトデータベースにアクセスしています。

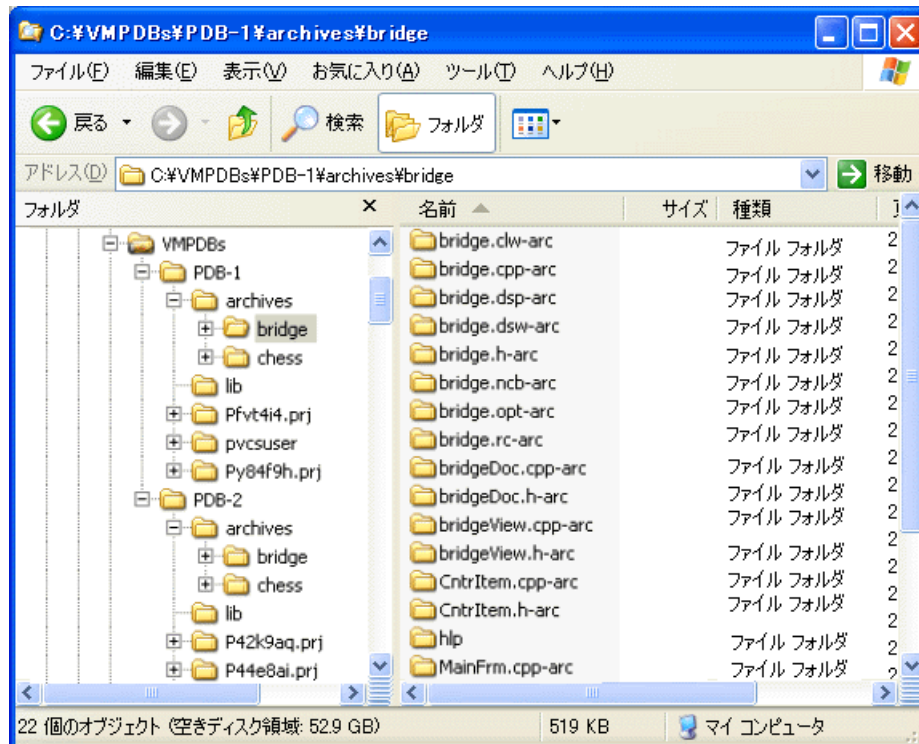
- S:¥VMPDBS¥PDB-1
 - S:¥VMPDBS¥PDB-2
- 1 リビジョンライブラリを作成する対象であるプロジェクトデータベースで `RFSSplitOnCreate` ディレクティブを有効にします。159 ページの『RFSSplitOnCreate ディレクティブの有効化』を参照してください。
 - 2 VSPLIT コマンドは、〈ユーティリティの使用を許可〉オプションによって有効になっている場合にのみ実行できます。142 ページの『File Server の設定』を参照してください。
 - 3 Version Manager クライアントシステムのコマンドラインから次のコマンドを実行します。
`vsplit -xo+eOutPut.txt -r S:¥VMPDBS`

VSPLIT コマンドは、Version Manager のインストール場所の `vm\OS\bin\admin` ディレクトリにあります。

`-xo+e` オプションによって操作の結果を分析することができます。162 ページの『VSPLIT 出力の分析』を参照してください。

クライアントを File Server に接続する方法については、168 ページの『File Server とともに使用するためのクライアントの設定』を参照してください。

次の図は、VSPLIT を実行した後のプロジェクトデータベースファイルの場所を示したものです。メタデータアーカイブは `C:\VMPDBs` の下のアーカイブ場所にあるのに対し、リビジョンは `C:\VMRvLb` の下にあります。これらの場所が File Server でどのようにマップされたかについては、150 ページの『例：既存のプロジェクトデータベースを移動せずに追加する場合』を参照してください。



VSPLIT 出力の分析

VSPLIT は、分割の前後に各アーカイブを調べます。破損したアーカイブあるいは非アーカイブファイルは分割しません。

各種非アーカイブファイル（アクセスコントロールデータベース、ジャーナルファイル、コンフィグレーションファイル）は変換失敗と表示されますが、これは予想どおりで問題ありません。出力を調べて失敗したアーカイブを確認してください。

各ファイルに対してエントリが追加されます。次の例は、無事分割されたアーカイブと、アーカイブではないため（予想通り）分割に失敗したジャーナルファイルを示します。

```
convert(S:\VMFS\PDBS\archives\test\env_filt.sed-arc) ok 733/1425 bytes (new/old)
vcheck(S:\VMFS\PDBS\archives\test\journal.vcs)=1 failed, not converting
```

VSPLIT 出力に予期せぬ失敗があった場合は、`pvcsfs.log` ファイルを確認します。このファイルに記録されているエラー、は根本的な問題を解決する手がかりになります。詳しくは、148 ページの『サーバログの表示』を参照してください。

アーカイブのエクスポート、インポート、移動、名前変更、および修正

VTRANSFER は、Version Manager File Server とともに使用できるコマンドラインユーティリティであり、以下を行うことを可能にします。

- 分割アーカイブなどのアーカイブのリビジョンデータを単一の *.zip ファイルとしてエクスポートします。これは、アーカイブを収集してパッケージ化し、トラブルシューティングの目的で Serena サポート担当者に送信できるようにする便利な方法です。
- 分割アーカイブなどのアーカイブのメタデータとリビジョンデータを単一の *.zip ファイルからインポートします。これは、アーカイブをトラブルシューティングするための変更をインポートする便利な方法です。
- アーカイブを同じサーバ上の異なる場所の間で移動します。
- アーカイブの名前を変更します。
- アーカイブが手動で移動または名前変更されている場合に、メタデータとリビジョンデータ（リビジョンライブラリ）の間の関係を修正します。
- アーカイブ、メタデータとリビジョンデータの両方を削除します。

構文

```
vtransfer [options] archivePath [secondaryPath]
```

各パラメータの意味は次のとおりです。

- *archivePath* は、作業中のアーカイブの場所です。
- *secondaryPath* は、*.zip ファイルの場所やアーカイブの新しい名前など、VTRANSFER の使用モードに特有なパラメータです。

オプション

VTRANSFER には、実行する操作の種類を決定する 4 つのモードオプションのほか、7 つの一般的なオプションがあります。

モードオプション

- d すべてのリビジョンデータとメタデータも含め、*archivePath* に指定されているアーカイブを削除します。
- f メタデータの場所をミラーする場所にリビジョンを移動することにより、メタデータとリビジョンデータの間関係を修正し、メタデータを更新します。このオプションは、メタデータが手動で移動または名前変更されている場合に有用です。
- i *secondaryPath* で指定した zip ファイルから、*archivePath* で指定したアーカイブの場所まで、アーカイブとそのリビジョンをインポートします。既存のアーカイブデータはすべて上書きされます。
- r *secondaryPath* で指定した名前にアーカイブまたはアーカイブのディレクトリを変更するか、ここで指定した場所に移動します。この操作は、*archivePath* がアーカイブではなくディレクトリを指定している場合は再帰的に実行されます。
- x *archivePath* で指定したアーカイブのメタデータとリビジョンデータをエクスポートし、*secondaryPath* で指定した単一の *.zip ファイルに書き込みます。*secondaryPath* に完全修飾パスが含まれない場合、ファイルはコマンドの実行元の場所へ書き込まれます。リビジョンは、File

Server 上にマップされたリビジョンパスと *archivePath* によって指定されるパスからエクスポートされます。

メタデータ内の指定した場所からリビジョンをエクスポートするには、**-m** オプションを併用します。これは、メタデータが手動で移動されている場合に便利です。

一般的なオプション

@ *@[list_file]*

このオプションにより、追加のコマンドラインオプションを記述した *list_file* を読み込みます。Version Manager は、指定したファイルから追加のオプションを読み込んでから、コマンドラインのそれ以外の部分を読み込みます。別のコマンドからの入力を転送する場合は、**@** オプションをパラメータを付けずに使用します。

ファイルの拡張子が *.GRP* である場合は、パスと拡張子を省略できます。Version Manager は、VCSDir ディレクティブで指定されたディレクトリ内でファイルを検索します。

-h コマンドに関するヘルプを表示します。このコマンドは、ほかのオプションを指定した場合でも、**-h** を処理した後で終了します。

-id *-id user_id: user_password*

Version Manager File Server の管理ユーティリティ内の該当するファイルサーバパスマップに関連付けられている、アクセスコントロールデータベースおよび / または LDAP サーバに対して認証する、ユーザ ID とパスワードを指定します。

注 **-id** オプションは、CLI 操作自体を行うために使用するユーザ ID には影響しません。CLI 操作のユーザ ID は通常の方法で決まります。

-m **-x** とともに使用し、メタデータ内の指定した場所からリビジョンをエクスポートします。これは、メタデータが手動で移動されている場合に便利です。

-xe *-xe file_name*

ステータス、プログラム、およびエラーメッセージを *file_name* に転送します。

-xo *-xo file_name*

標準出力を *file_name* にリダイレクトします。

例

以下の例では、アーカイブは、クライアント名である *S:¥VMFS¥PDBs* を使用して、Version Manager File Server にマップされています。

例 1 : アーカイブを Zip ファイルにエクスポートする

この例は、バージョン管理ファイル *Hello.txt* と関連付けられたリビジョンとメタデータを取得して、それを *Hello.zip* という名前のファイルに格納するものです。これは、単一のバージョン管理ファイルを表すさまざまなファイルを収集し、パッケージ化する便利な方法です。

```
vtransfer -x S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc Hello.zip
```

例 2 : メタデータに格納されたリビジョンパスを使用してアーカイブを Zip ファイルにエクスポートする

この例は例 1 と似ていますが、File Server 上にマップされたリビジョンパスと *archivePath* によって指定されるパスではなく、メタデータで指定された場所 (-m) からリビジョンが取得される点が異なります。これは、メタデータが手動で移動または名前変更されている場合に便利です。

```
vtransfer -x -m S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc Hello.zip
```

例 -3 : アーカイブを Zip ファイルからインポートする

この例は、例 1 と 2 で生成されたファイルなど、エクスポートファイルからメタデータとリビジョンをインポートするものです。既存のアーカイブ情報はすべて上書きされます。

```
vtransfer -i S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc Hello.zip
```

例 4 : アーカイブを移動する

この例は、バージョン管理ファイルに関連付けられたメタデータとリビジョンを File Server 上の新しい場所に移動するものです。

```
vtransfer -r S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc
S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Boneyard¥Hello.txt-arc
```

例 5 : アーカイブの名前を変更する

この例は、File Server 上のバージョン管理ファイルの名前を変更して、メタデータとリビジョンをそれに合わせて更新するものです。

```
vtransfer -r S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txv
S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc
```

例 6 : アーカイブを修正する

この例は、メタデータの場所をミラーする場所にリビジョンを移動することにより、メタデータとリビジョンデータの間の関係を修正し、メタデータを更新するものです。これは、メタデータが手動で移動または名前変更されている場合に有用です。

```
vtransfer -f S:¥VMFS¥PDBs¥PDB-1¥archives¥Project-1¥Hello.txt-arc
```

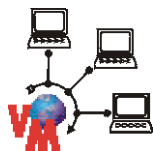
クロスプラットフォーム環境における File Server の使用

UNIX と Windows の両システムから同じアーカイブにアクセスするには、作成と設定に関する特別な手順を実行して、クロスプラットフォーム環境でのアクセスで問題が起きないようにする必要があります。

作成 プロジェクトデータベースを UNIX サーバに配置し、それに Windows クライアントからアクセスする場合は、UNC パス (¥¥myserver) または Windows スタイルのドライブマッピング (S:¥myserver) を使用して、Windows クライアントプロジェクトデータベースを作成する必要があります。UNIX スタイルのパスを使用して作成されたプロジェクトデータベースには、クロスプラットフォーム環境でアクセスすることはできません。

設定 アーカイブにアクセスする場合は以下が必要です。

- 直接アクセスする場合も、File Server 経由でアクセスする場合も、クロスプラットフォーム環境に関するセクションで説明されているすべての手順を実行する必要があります。
- File Server 経由でアクセスする場合は、以下の手順だけを実行する必要があります。
 - EOL シーケンスを変換する。
 - ユーザ ID の大文字と小文字を区別しないようにする。
 - ファイル名とディレクトリ名の大文字と小文字を一致させるようにする



注 説明の中で、これらの手順には左に示す図が付けられています。

これらの手順を実行する場合の参照情報：

- デスクトップクライアントを使用する場合は、[41 ページの『クロスプラットフォーム環境での Version Manager のセットアップ』](#)を参照してください。
- コマンドラインを使用する場合は、[56 ページの『クロスプラットフォーム環境』](#)を参照してください。

セキュリティに関する特記事項

デフォルトで、Version Manager File Server は、セキュリティとパフォーマンスのニーズを均衡させるように設定されています。File Server を状況に合わせて最適化するには、使用環境と用途を入念に評価することをお勧めします。以下の推奨事項に目を通してください。

一般的な推奨事項

- まず、どんな場合でも以下の設定を推奨します。
 - 〈認証要求〉オプションを有効にする（デフォルト設定）。
 - 〈リモートの設定を許可〉オプションを無効にする（デフォルト設定）。
 - 〈ユーティリティの使用を許可〉オプションを「ローカル専用」に設定する（デフォルト設定）。
 - パスマップのアクセスコントロールデータベースを有効にする。[167 ページの『パスマップのデフォルトアクセスコントロールデータベースの作成』](#)を参照してください。
 - Version Manager File Server をホストするシステムへのアクセスを制限する。ファイルサーバホストで実行する Version Manager クライアントは特定のメタデータ読み取り操作に対してファイルサーバセキュリティをバイパスすることができます。ただし、すべての書き込みおよびリビジョン操作にはセキュリティが適用されます。

インターネットに特有な推奨事項

- ファイアウォール外部のユーザが Web クライアントにアクセスできるようにするには
 - Version Manager File Server をファイアウォールの外にある別のシステム上に配置する（Version Manager Web サーバをファイアウォールの外に置きます）。
 - SSL サポートを Version Manager File Server で有効にする。[167 ページの『File Server における Secure Socket Layer の有効化』](#)を参照してください。

詳細情報 Version Manager セキュリティの設定に関する詳細は、次の表を参照してください。

参照内容	参照
ファイルサーバセキュリティオプションの設定	142 ページの『File Server の設定』
パスマップセキュリティオプション	141 ページの『パスマップセキュリティオプションの設定』
Version Manager セキュリティの設定	201 ページの第 6 章『セキュリティの使用』

パスマップのデフォルトアクセスコントロールデータベースの作成

パスマップのアクセスを制御するアクセスコントロールデータベースを指定することができます。いったんアクセスコントロールデータベースを指定すると、すべてのアクセスがそれを使用して確認されます。有効なユーザ ID がないとパスマップのプロジェクトデータベースを作成することができません。存在しないアクセスコントロールデータベースを指定すると、誰もパスマップにアクセスすることができなくなります。

パスマップセキュリティのセットアップを容易にするには

- 1 ファイルサーバパスマップ上にないダミープロジェクトデータベースを作成します。
- 2 ダミープロジェクトデータベースのアクセスコントロールデータベースを設定します。
- 3 アクセスコントロールデータベースをファイルサーバからアクセス可能な場所へコピーします。
- 4 オプション表示部の〈アクセスコントロールデータベース〉フィールドにアクセスコントロールデータベースの場所を指定します。指定したアクセスコントロールデータベースがファイルサーバ上で作成されるすべての新しいパスマップに関連付けられます。

File Server における Secure Socket Layer の有効化

Version Manager Application Server で Secure Socket Layer (SSL) を有効にすると、セキュリティを強化することができます。



注 Version Manager Application Server で SSL を有効にすると、そこで実行されるすべての Version Manager アプリケーション (Version Manager File Server、Version Manager Web Server、および WebDAV) が影響を受けます。

以下の手順に従うと、SSL を有効にすることができますが、SSL に関するすべての問題が解決されるわけではありません。SSL の有効化について詳しくは、次の文書を参照してください。

`Install_Dir¥serena¥vm¥common¥tomcat¥webapps¥tomcat-docs¥ssl-howto.html`



注 `ssl-howto.html` ファイルの「Download and Install JSSE」の部分は無視してください。JSSE はすでにインストールされています。

SSL を有効にするには

- 1 コマンドプロンプトから次のコマンドを実行します (UNIX ではパス区切りをスラッシュに変えます)。

```
Install_Dir¥serena¥vm¥common¥jre¥05¥bin¥keytool -genkey -alias tomcat -keyalg RSA
```

- 2 プロンプトが表示されたら、キースタアのパスワードを入力します。デフォルトは `changeit` です。
- 3 プロンプトが表示されたら、サーバへのアクセス時にユーザに対して表示する情報（会社名など）を入力します。
- 4 プロンプトが表示されたら、キー自体のパスワードを入力します。このパスワードは、上記の手順で入力したキースタアのパスワードと同じである必要があります。
- 5 次のファイルをテキストエディタで開きます（UNIX ではパス区切りをスラッシュに変えます）。
`Install_Dir¥Serena¥vm¥common¥tomcat¥conf¥server.xml`

- 6 ファイルで次のセクションを見つけます。

```
<!-- Define a SSL Coyote HTTP/1.1 Connector on port 8443 -->
<!--
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
  port="8443" minProcessors="5" maxProcessors="75"
  enableLookups="true" acceptCount="100" debug="0" scheme="https"
  secure="true" useURIVValidationHack="false"
  disableUploadTimeout="true">
<Factory
  className="org.apache.coyote.tomcat4.CoyoteServerSocketFactory"
  clientAuth="false" protocol="TLS" />
</Connector>
-->
```

- 7 テキストブロックを囲んでいるコメントの開始タグと終了タグを削除します（セクション見出しは残します）。こうすることにより、SSL がポート 8443 で有効になります。別のポートを指定するには、`ssl-howto.html` ファイルを参照してください。



注 デフォルトの HTTP ポートである 8080 はまだ機能します。必要に応じて、ポート 8080 を有効にするテキストブロックをコメントアウトするか、このポートへのアクセスを制限してください。

- 8 パフォーマンスを向上させるには、`enableLookups` を `true` から `false` に変更します。
- 9 ファイルを保存し、Version Manager Application Server を再起動します。
- 10 クライアントが `https:` と SSL が有効になったポートを使用するように設定します。[168 ページの『File Server とともに使用するためのクライアントの設定』](#)を参照してください。

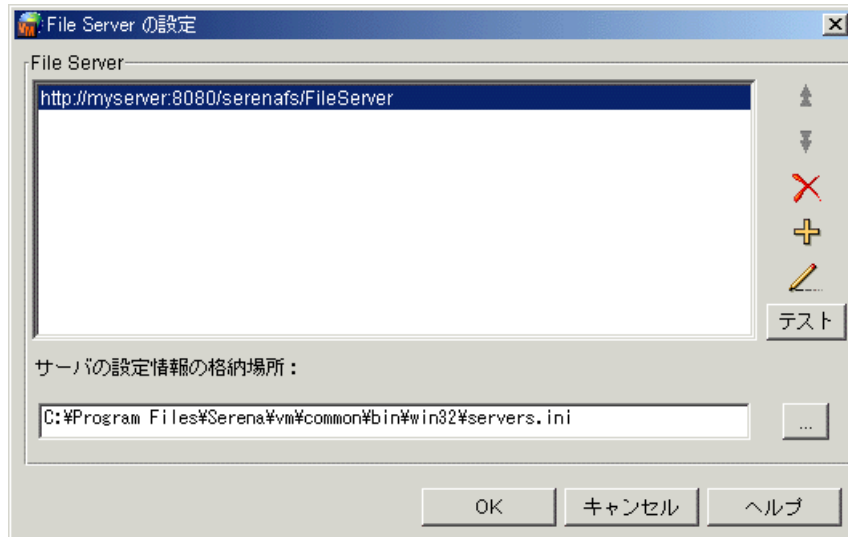
File Server とともに使用するためのクライアントの設定

File Server の場所をすべての Version Manager クライアントが認識できるようにするには、サーバコンフィグレーションファイルに Version Manager File Server へのパスを入力する必要があります。サーバコンフィグレーションファイルをネットワーク上の場所に保存する場合、新しいサーバコンフィグレーションファイルを各ユーザのシステムで作成するのではなく、各クライアントシステムにその場所を参照させることができます。

デスクトップクライアントのインストール時に File Server へのアクセスを設定する方法

File Server で使用するようクライアントを設定するには

- 1 デスクトップクライアントを起動します。
- 2 [管理] - [File Servers] を選択します。[File Server の設定] ダイアログボックスが表示されます。



- 3 以下のうち必要な作業を行います。



- 選択したサーバの〈File Servers〉リストにおける位置を変更するには、上下の矢印ボタンをクリックします。このリストの一番上から順に自動的に検索されます。



- 選択したサーバをリストから削除するには、【削除】ボタンをクリックします。



- リストにサーバを追加するには、【追加】ボタンをクリックします。[Add File Server URL] ダイアログボックスが表示されます。

- a サーバをホストするシステムの名前を、*MyServer* などのように入力します。オプションで、*MyServer:Port* などのようにポートを指定することもできます。デフォルトで、ポート 8080 が使用されます (Web サーバを Version Manager Web Server Application とともに使用するよう設定してある場合は、そのポート番号を使用することができます)。

デフォルトで、エントリの頭には `http://` が付けられます。SSL サーバを指定するには、`https://MyServer` と入力します。

ファイルサーバにローカルなクライアントが使用するポートを設定するには、ファイルサーバをホストしているシステムで [File Server の設定] ダイアログボックスを設定します。



重要! ファイルサーバと Web サーバが同じシステム上にあり、同時実行ユーザが 51 人以上いる場合、ファイルサーバのローカルクライアントが使用するポートを 8090 に設定します。そうしないとサーバがダウンする可能性があります。デフォルトでは、サーバのローカルクライアントはポート 8080 を使用します。

b 【OK】 をクリックします。



- 選択したサーバの URL を編集するには、【編集】 ボタンをクリックします。[Edit File Server URL] ダイアログボックスが表示されます。

a <名前> テキストボックスの URL を編集します。

b 【OK】 をクリックします。

- 選択したサーバへの接続をテストするには、【テスト】 ボタンをクリックします。

- 4 <サーバの設定情報を格納するファイルの名前> フィールドに、サーバ設定情報を格納するファイルのパスと名前 (*.ini) を入力するか、デフォルトの場所をそのまま使用します。

<File Servers> リストの値を自動的に挿入する

すべてのユーザに対して設定作業を簡易化するには、サーバのリストを設定し、すべてのユーザがそれにアクセスできるネットワーク上の場所にコンフィグレーションファイルを配置します。ユーザが誤ってファイルを修正することのないように、この場所は読み取り専用にします。こうすることにより、ユーザは、コンフィグレーションファイルの共有場所を指定するだけで、各自の <File Servers> リストの値を自動的に挿入することができます。

ワークステーションインストールを行う場合は、デフォルトの場所は、ネットワーク上の共有場所に自動的に設定されます。

- 5 【OK】 をクリックします。

IDE クライアントのみのインストール時に File Server へのアクセスを設定する方法



注 servers.ini ファイルの場所が islv.ini ファイルで指定されていない場合、File Server に初めてアクセスしようとする、ローカルな servers.ini ファイルが IDE クライアントによって作成されます。すべてのユーザにネットワーク上の 1 つの servers.ini ファイルにアクセスさせる場合、171 ページの『デスクトップクライアントがインストールされていない場合に Servers.ini ファイルの場所を指定する方法』を参照してください。



注 この手順は、Eclipse および .NET へのリッチ統合には適用されません。それらのインターフェイスへのファイルサーバアクセスを有効にするにはデスクトップクライアントまたは手動方法を使用してください。

File Server のアクセスを IDE クライアントから設定するには

- 1 取得やチェックアウトなど、プロジェクトデータベースを開くことができる IDE クライアントの任意のダイアログボックスで、【データベースを開く】 ボタンをクリックします。[File Server の選択] ダイアログボックスが表示されます。

サーバを追加 / 編集する

- 2 新しいサーバを追加するには、「File Server」列で空白のセルをダブルクリックします。既存のサーバエントリを編集するには、そのサーバのエントリをダブルクリックします。

- 3 サーバをホストするシステムの名前を、MyServer などのように入力します。オプションで、MyServer:Port などのようにポート番号を指定することもできます。デフォルトで、ポート 8080 が使用されます。

デフォルトで、エントリの頭には http:// が付けられます。SSL サーバを指定するには、https://MyServer と入力します。

サーバを削除する

- 4 サーバエントリを削除するには、「File Server」列でサーバを選択し、DELETE キーを押します。

- | | |
|-------------|--|
| サーバの順序を変更する | 5 「File Server」リストにおけるサーバの位置を変更するには、それを必要な位置までドラッグしてドロップします。このリストの一番上から順に自動的に検索されます。 |
| 変更を保存する | 6 servers.ini ファイルに加えた変更を保存するには、【OK】ボタンをクリックします。 |
| サーバをテストする | 7 「File Server」列に表示されたサーバへの接続をテストするには、【接続テスト】ボタンをクリックします。「ステータス」列で各サーバのステータスが更新されます。 |

デスクトップクライアントがインストールされていない場合に Servers.ini ファイルの場所を指定する方法

各ワークステーションに関する Version Manager の islv.ini ファイル (Windows) または .islvrc ファイル (UNIX) を直接編集して、servers.ini ファイルの場所を指定することができます。これは、以下の場合に行います。

- コマンドラインインターフェイスしかインストールされていない場合。
- IDE クライアントしかインストールされておらず、すべてのユーザがネットワーク上の単一の servers.ini ファイルにアクセスできるようにする場合。



注 servers.ini ファイルの場所が islv.ini ファイルで指定されていない場合、File Server に初めてアクセスしようとすると、ローカルな servers.ini ファイルが IDE クライアントによって作成されます。

クライアントの islv.ini ファイルを手動で更新するには

- 1 islv.ini ファイル (Windows) または .islvrc ファイル (UNIX) をテキストエディタで開きます。これらのファイルは、Windows のルートディレクトリ (通常は C:¥WINNT) または UNIX のホームディレクトリにあります。
- 2 [PVCSGUI_6.5] セクションにある pvcs.fileserver.path= エントリを見つけます。これは、Windows の場合はデフォルトで以下ようになります。

```
pvcs.fileserver.path=C:¥Program Files¥Serena¥vm¥common¥bin¥win32¥servers.ini
```

UNIX の場合は以下ようになります。

```
pvcs.fileserver.path=/usr/serena/vm/common/bin/05/servers.ini
```

- 3 設定した servers.ini ファイルの場所を反映するようにパスを修正します。



注 servers.ini ファイルは、デスクトップクライアントからも IDE クライアントからも設定できます。

- 4 ファイルを保存します。



ヒント ワークステーションインストール機能を使用してワークステーションのセットアップを自動化する予定の場合は、ワークステーションのインストール時に islv.ini または .islvrc の各ファイルを修正することにより、時間を節約することができます。ワークステーションインストールの詳細については、『Version Manager インストールガイド』を参照してください。

Version Manager クライアントから File Server にアクセスする方法

ユーザのシステムが設定されたら、ユーザは、サーバ対応のプロジェクトデータベースに以下からアクセスすることができます。

- **デスクトップクライアント** : [ファイル] - [プロジェクトデータベースを開く] を選択し、(検索先) リストから「File Servers」を選択します。
- **IDE クライアント** : 取得やチェックアウトなど、プロジェクトデータベースを開くことができる任意のダイアログボックスを使用するか、【データベースを開く】ボタンをクリックします。
- **コマンドライン (CLI および PCLI)** : アーカイブ、コンフィグレーションファイル、またはプロジェクトデータベースのパスをクライアント名 (Version Manager File Server の管理ユーティリティの [パスマップ] ウィンドウで定義) に対する相対パスで指定します。次にその例を示します。

```
¥¥ClientName¥Project-1¥Foo.txt-arc
```



注 クライアント名にアクセスコントロールデータベースまたは LDAP 認証が関連付けられている場合、`-id` オプションを使用するか環境変数を設定して確認用のユーザ ID とパスワードを与える必要があります (`-iduser_id:user_password`)。『Version Manager コマンドラインリファレンスガイド』を参照してください。

- **プロジェクトコマンドライン (PCLI)** : アーカイブ、コンフィグレーションファイル、またはプロジェクトデータベースのパスをクライアント名 (Version Manager File Server の管理ユーティリティの [パスマップ] ウィンドウで定義) に対する相対パスで指定します。次にその例を示します。

```
¥¥ClientName¥Project-1¥Foo.txt-arc
```

- **Web クライアント** : Version Manager File Server 上にあるプロジェクトデータベースにアクセスするよう Web サーバを設定します。これを行うには、サブレットのセットアップ時にプロジェクトデータベースの場所をクライアント名に対する相対パスで指定します。Web クライアント自体のセットアップ作業は一切不要です。



重要! ファイルサーバと Web サーバが同じシステム上にあり、同時実行ユーザが 51 人以上いる場合、ファイルサーバのローカルクライアントが使用するポートを 8090 に設定します。そうしないとサーバがダウンする可能性があります。デフォルトでは、サーバのローカルクライアントはポート 8080 を使用します。169 ページの『デスクトップクライアントのインストール時に File Server へのアクセスを設定する方法』を参照してください。

第 5 章

環境の管理

はじめに	174
ワークスペースの使用	174
カスタムツールの追加	182
既存アーカイブの属性の変更	185
アーカイブの移動	192
プロジェクトとプロジェクトデータベースのコピー	195

はじめに

この章では、Serena ChangeMan Version Manager の環境を管理するために行う作業について説明します。管理者として行う可能性のあるものとしては、パブリックワークスペースの定義、ツールバーおよび [ツール] メニューへのカスタムツールの追加、既存アーカイブの属性の変更、およびアーカイブの場所の移動などがあります。

ワークスペースの使用

ワークスペースとは、プロジェクトデータベースに対して定義された一連の作業設定を意味します。具体的には、ワークスペースには次のものが保存されます。

- プロジェクトデータベース、およびそこに含まれるプロジェクト、サブプロジェクト、およびバージョン管理ファイルに定義された作業ファイルの場所。作業ファイルの場所とは、ファイルのチェックイン元およびチェックアウト先となるディレクトリで、プロジェクトデータベースの作成、プロジェクトの作成、または作業ファイルの追加を行うときに定義されます。
- リビジョン番号もバージョンラベルも指定されていない場合にアクション（チェックアウトなどの）の対象となるデフォルトのリビジョン。これは通常、浮動ラベルが設定されたリビジョンで、特定のブランチのチップが選択されます。
- 自動ブランチが行われる場合に使用されるベースバージョンおよびブランチバージョン。
- デフォルトプロモーショングループ。ただし、プロモーションモデルが設定されている場合に限り。デフォルトプロモーショングループは、プロモーションモデルの最下位のプロモーショングループです。

ワークスペースは、プロジェクトデータベースに対してデスクトップクライアントでのみ作成できません。複数のワークスペースを作成できますが、同時に設定できるのは、1 つのプロジェクトデータベースのみです。

ワークスペースにはプロジェクトデータベースのみが関連付けられていますが、これにはデータベース内のすべてのプロジェクトとサブプロジェクトの設定が含まれます。たとえば、プロジェクトデータベースに対して定義された作業ファイルの場所、およびデータベース内のすべてのプロジェクトはワークスペースに保存されます。プロジェクトデータベースに関して設定できるワークスペースは一度に 1 つだけですが、データベース中の各プロジェクトには、そのワークスペース内に別々に作業ファイルの場所を定義できます。

ワークスペースのタイプ

ワークスペースにはパブリックとプライベートの 2 種類があります。パブリックワークスペースの設定は、そのワークスペースを使用する全員に影響します。プライベートワークスペースの設定は、それを作成して設定したユーザにのみ影響します。プライベートワークスペースはほかのユーザとは共有できません。

Version Manager では、プロジェクトデータベースが作成されるときに、「ルートワークスペース」というデフォルトのワークスペースが作成されます。1 つのプロジェクトデータベースには、1 つの「ルートワークスペース」のみが定義可能です。ルートワークスペースに定義される初期設定は次のとおりです。

- データベースを作成するときにプロジェクトデータベースに対して設定する作業ファイルの場所。

- プロジェクトデータベースに関連付けられたマスターコンフィグレーションファイルに、デフォルトのリビジョンが定義されている場合に、アクションに使用されるデフォルトのリビジョン。
- プロジェクトデータベースに関連付けられたマスターコンフィグレーションファイルに、自動ブランチが設定されている場合の自動ブランチ。初期設定は、マスターコンフィグレーションファイルから取得されます。

初期設定に含まれないものに、デフォルトプロモーショングループがあり、これは、プロモーションモデルが設定されている場合にのみ有効です。デフォルトプロモーショングループは、プロモーションモデルの最下位のプロモーショングループです。デフォルトでは、リビジョンのチェックアウト、リビジョンのロック、および作業ファイルの追加時に、指定したプロモーショングループがリビジョンに割り当てられます。デフォルトプロモーショングループを定義すると、ユーザがこれらのアクションを行うときに、割り当てるべき最下位のプロモーショングループを指定する必要がなくなります。

ルートワークスペースは削除したり、名称を変更することはできません。ただし、必要な権限を与えられていれば、どのユーザもルートワークスペース内に定義された設定を編集できます。[238 ページ](#)の権限の表を参照して、権限が正しく付与されるようにしてください。

プロジェクトをプロジェクトデータベースに追加する場合は、それぞれのプロジェクトに対するワークスペースの設定がルートワークスペースの定義に追加されます。

ルートワークスペースが変更されると、それを使用するすべてのユーザに影響を及ぼします。このため、同じプロジェクトデータベースに複数のユーザがアクセスする場合は、プライベートワークスペースの使用をお勧めします。一般的には、管理者がいくつかのパブリックワークスペースを作成し、特定のユーザグループ（開発、製造、品質保証など）の作業環境を定義します。管理者が、ユーザによるパブリックワークスペースの変更、削除、および名称変更を禁止しても、ユーザは、プライベートワークスペースを作成し、ローカルドライブで作業を行うことができます。

ワークスペースの階層

ワークスペース階層の最上層はルートワークスペースです。次に示す例では、ルートワークスペースの下に Dev、Production、QA、および UNIX はすべて子ワークスペースです。



ルートワークスペースの初期設定は、次のとおりに定義されています。

設定	当初の設定
作業ファイルの場所	プロジェクトデータベースとプロジェクトが作成されるとき、作業ファイルの場所。
デフォルトバージョン ブランチバージョン ベースバージョン	プロジェクトデータベースに関連付けられたマスターコンフィグレーションファイル、および（存在する場合は）プロジェクトに関連付けられたプロジェクトコンフィグレーションファイル。
デフォルトプロモーショングループ	この値は、自動的に設定されません。

ワークスペースを作成するときは、新しいワークスペースが設定を引き継ぐワークスペースを選択する必要があります。ここで選択したワークスペースは、親ワークスペースと呼ばれます。新しいワークスペースの設定は自由に値を変更することができ、親ワークスペースから継承された値に優先して

使用されます。値を変更しない場合は、親ワークスペースのものと同じ値が使用されます。後でその親ワークスペースの設定を変更した場合は、その親ワークスペースから作成されたワークスペースの設定も変更されます。

たとえば、新しいワークスペースを作成し、引き継いだ作業ファイルの場所、Z:¥PRJDB¥WORK を変更しなかった場合は、後でこの親ワークスペースの作業ファイルの場所を Y:¥PRJDB¥WORK に変更すれば、子ワークスペースの作業ファイルの場所も Y:¥PRJDB¥WORK に変更されます。

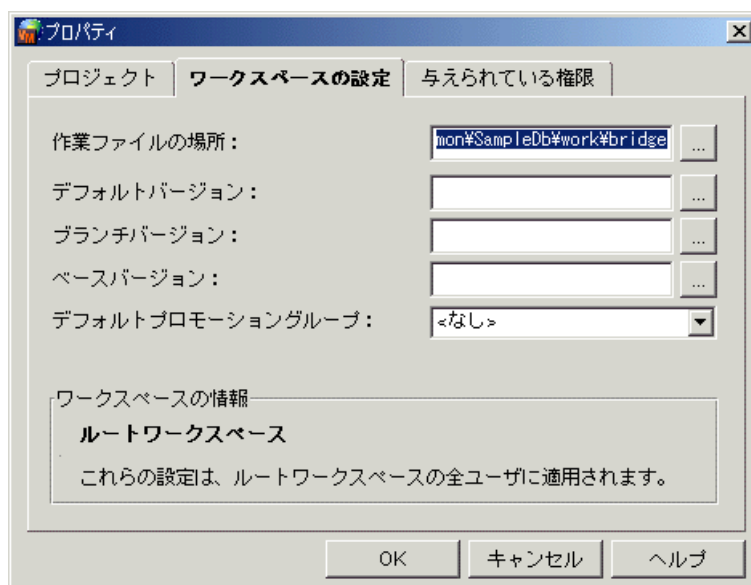
新しく作成されたワークスペースの初期値を一度変更すると、親ワークスペースの値を変更しても、子ワークスペースは影響を受けなくなります。たとえば、新しいワークスペースを作成し、継承した作業ファイルの場所を変更した後に親ワークスペースの作業ファイルの場所を変更しても、子ワークスペースの作業ファイルの場所は変更されません。

フィールドから現在の値を削除（フィールドを空白に）して、最初に継承した値に戻すことが可能です。

ワークスペースの設定の表示

ワークスペースの設定をチェックする必要が生じることがあります。この場合は、次の手順を実行します。

- 1 プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [ファイル] - [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示され、[プロジェクトデータベース] または [プロジェクト] タブがアクティブになっています。
- 3 [ワークスペースの設定] タブを選択します。このタブにはワークスペース設定のリストが表示されます。



注 プロモーションモデルがデータベースに対して定義されていない場合、〈デフォルトプロモーショングループ〉フィールドは [ワークスペースの設定] タブには表示されません。

- 4 ワークスペース設定を確認したら、【OK】をクリックします。ワークスペース中の作業ファイルの場所を表示または、編集するには、[ファイル] - [作業ファイルの場所の設定] を選択するほうが簡単です。

パブリックワークスペースの使用

管理者はすべてのユーザに影響を与えるパブリックワークスペースに対して作業を行います。プライベートワークスペースは通常、ユーザによって作成され、管理者は作成しません。プライベートワークスペースの使用については『Serena ChangeMan Version Manager ユーザガイド』を参照してください。

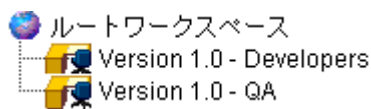
パブリックワークスペースは、次のような場合には特に便利です。

- 品質管理や開発など、異なるグループにより、異なるローカルまたはネットワークドライブ上で、同一の作業ファイルの場所が使用されている場合。
- 品質管理や開発、製造など、グループごとに同一ドライブ上に異なる作業ファイルの場所が設定されている場合。

異なるネットワークドライブが使用される場合

最初の場合においては、プロジェクトデータベース中のすべてのプロジェクトとサブプロジェクトについて、相対的な作業ファイルの場所を使用してルートワークスペースを定義します。プロジェクトデータベースの作業ファイルの場所は、`c:\¥prjdbl¥work` などの絶対パスであることが必要です。次にルートワークスペースから `class.lib` などのプロジェクトについて相対的な作業ファイルの場所を定義し、このプロジェクトの作業ファイルの場所は、`c:\¥prjdbl¥work¥class.lib` となります。

次に、「Version 1.0 - Developers」と「Version 1.0 - QA」の2つのパブリックワークスペースを作成します。

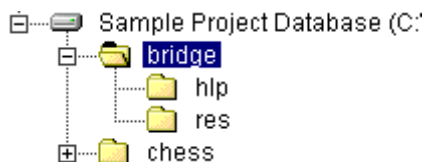


各パブリックワークスペースは、ルートワークスペースで定義された作業ファイルの場所を引き継ぎます。各パブリックワークスペースについては、`Z:\¥Version_1.0` と `Y:\¥Version_1.0` など、異なる作業ファイルの場所を絶対パスとして定義します。これらの絶対パスによる作業ファイルの場所は、各グループが実際に作業を行うドライブに一致します。相対的な場所（たとえば `class.lib`）は、プロジェクトデータベースの絶対パスによる作業ファイルの場所に追加して、たとえば `Z:\¥Version_1.0¥class.lib` が得られるため、プロジェクトの相対的な作業ファイルの場所を変更する必要はありません。

これらのワークスペースを作成することにより、品質管理（QA）の作業環境と、開発者（Developer）の作業環境を簡単かつ迅速に切り換えられます。

同一ドライブが使用される場合

異なる部署のためにそれぞれの作業ファイルの場所を同一ドライブ上に定義するには、プロジェクトデータベースに対しては絶対パスによる作業ファイルの場所、そのプロジェクトデータベース中のすべてのプロジェクトとサブプロジェクトに関しては相対パスによる作業ファイルの場所を使用してルートワークスペースを定義します。



異なる作業ファイルの場所を定義するには、部署ごとにパブリックワークスペースを作成します。この新しいパブリックワークスペースは作業ファイルの場所をルートワークスペースから引き継いだ

め、プロジェクトデータベース中のプロジェクトとサブプロジェクトのそれぞれの作業ファイルの場所に対し、部署のディレクトリを追加するだけで済みます。

たとえば Windows では次のように定義されます。

ワークスペース	作業ファイルの場所
ルート	Bridge プロジェクトを含むプロジェクトデータベース、C:¥sample¥work
ルート	Bridge プロジェクト、C:¥sample¥work¥bridge
QA	Bridge プロジェクト、C:¥sample¥work¥bridge¥qa
Dev	Bridge プロジェクト、C:¥sample¥work¥bridge¥dev
Production	Bridge プロジェクト、C:¥sample¥work¥bridge¥prod

この例では、QA のワークスペースについてルートワークスペースから継承した作業ファイルの場所の最後に ¥qa を追加します。完全な作業ファイルの場所は C:¥sample¥work¥bridge¥qa となります。ここで bridge¥qa は相対パスによる作業ファイルの場所になります。階層において、Bridge プロジェクトのすぐ上のプロジェクト（ここではプロジェクトデータベース）に関する作業ファイルの場所を変更した場合は、Bridge プロジェクトの作業ファイルの場所も変更されます。たとえば、プロジェクトデータベースの作業ファイルの場所を D:¥sample¥test¥work¥source に変更した場合は、Bridge プロジェクトの作業ファイルの場所が D:¥sample¥test¥work¥source¥bridge¥qa に変更されます。

パブリックワークスペースの作成、名称変更、および削除の制限

ユーザに SuperUser および Unlimited のどちらの権限も付与しないことにより、ユーザによるパブリックワークスペースの作成、名称変更、および削除を制限することができます。この 2 つの権限セットにより、これらのパブリックワークスペースに対するアクションの実行がユーザに許可されます。パブリックワークスペースを削除、名称変更、および削除できるユーザは、ほかのユーザの作業に支障をきたす可能性があります。



注 ユーザがプロジェクトルートにログインすると、そのアクセス権がチェックされ、パブリックワークスペースでの作業が許可されているかどうか判断されます。ご使用の環境に、コンフィグレーションファイルおよび、プロジェクトルートのサブプロジェクトについてのアクセスコントロールデータベースが存在する場合は、パブリックワークスペースでの作業が許可されているかどうか判断するときに、サブプロジェクトにログインしているユーザの権限は使用されません。

次にその例を示します。

- あるユーザが、ほかのユーザのデフォルトワークスペースであるパブリックワークスペースの名前を変更すると、ルートワークスペースが、これらのユーザのデフォルトワークスペースになります。
- あるユーザが、ほかのユーザが使用するパブリックワークスペースを削除すると、そのワークスペースは使用不可能になり、ほかのユーザの作業に支障をきたします。
- あるユーザがパブリックワークスペースを削除し、同じ名前のパブリックワークスペースを新規作成すると、そのワークスペースは使用可能ですが、設定が以前とは異なっている可能性があります。

ユーザに対する権限の割り当ての詳細については、[201 ページの『セキュリティの使用』](#)を参照してください。

また、特定の権限をユーザに禁止して、パブリックワークスペースで定義されている設定の、ユーザによる変更を制限することもできます。[238 ページの権限の表](#)を参照してください。

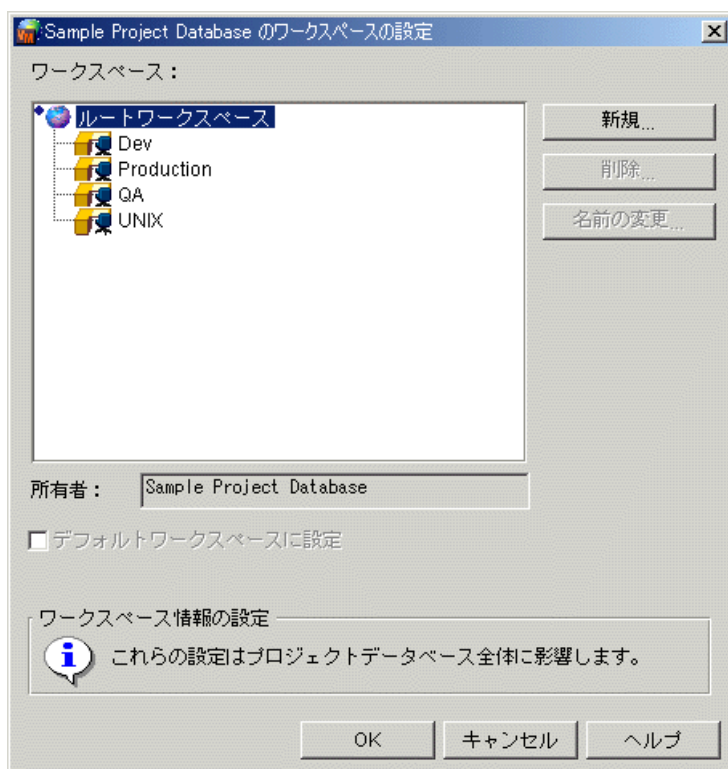
パブリックワークスペースの作成

管理者は、すべてのユーザに対して影響するパブリックワークスペースを作成します。プライベートワークスペースはユーザによって作成され、管理者によっては作成されません。プライベートワークスペースの作成については『Serena ChangeMan Version Manager ユーザガイド』を参照してください。

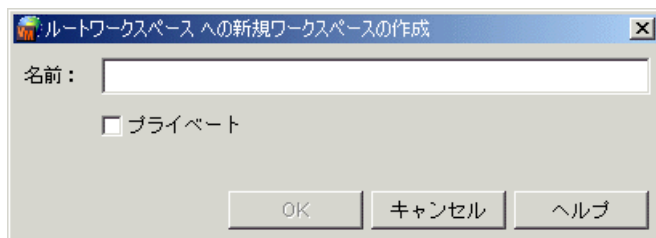
パブリックワークスペースを作成するときには、作成する新しいワークスペースが設定を受け継ぐパブリックワークスペースを 1 つ選択します。175 ページの『ワークスペースの階層』を参照してください。

パブリックワークスペースを作成するには

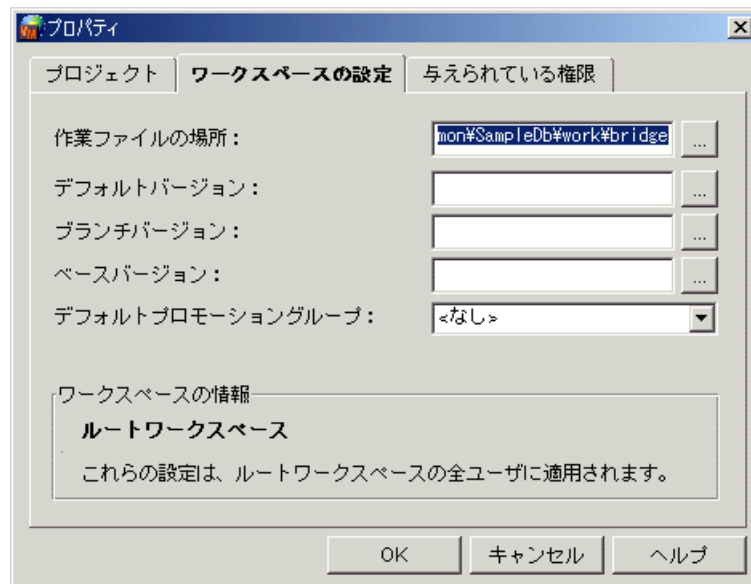
- 1 パブリックワークスペースを作成するプロジェクトデータベースを選択します。
- 2 [ファイル] - [ワークスペースの設定] を選択します。[ワークスペースの設定] ダイアログボックスが表示されます。



- 3 新しいワークスペースが設定を引き継ぐパブリックワークスペースを選択します。パブリックワークスペースはプライベートワークスペースからは設定を引き継ぐことはできません。
- 4 【新規作成】をクリックします。[新規ワークスペースを作成] ダイアログボックスが表示されます。



- 5 ワークスペースの一意の名称を入力します。親ワークスペースが異なる場合を除き、複数のワークスペースが同じ名前を持つことはできません。
- 6 【OK】をクリックします。〔ワークスペースの設定〕ダイアログボックスが表示され、新しいワークスペースが選択されています。この新しいワークスペースを現在のワークスペースとして指定するために、選択をそのままにしておきます。現在のワークスペースとして指定すると、その設定を定義できます。
- 7 【OK】をクリックして〔ワークスペースの設定〕ダイアログボックスを閉じます。これで新しいワークスペースが現在のワークスペースとして指定されます。
- 8 以下を実行して、新しいワークスペースの設定を定義します。
 - a [ファイル] – [プロパティ] を選択します。〔プロパティ〕ダイアログボックスが表示され、〔プロジェクトデータベース〕または〔プロジェクト〕タブがアクティブになっています。
 - b [ワークスペースの設定] タブを選択します。このタブにはワークスペース設定のリストが表示されます。



- c 次の作業を行って設定を変更します。
 - 作業ファイルの場所を変更します。相対パスによる作業ファイルの場所を定義するには、作業ファイルの最初の場所にディレクトリ名を追加するか、qaなどのディレクトリ名のみを入力します。ディレクトリ名のみを入力した場合は、このディレクトリが親プロジェクトの作業ファイルの場所ディレクトリに追加されます。

プロジェクトデータベースに対しては相対パスによる作業ファイルの場所は指定できません。UNIXでは、作業ファイルの場所のパスのルートに \$HOME を含めることができます (たとえば、\$HOME/work は /usr/cherylc/work に展開できます)。Version Manager は作業ファイルの場所を判断するときに、HOME 環境変数の値を置き換えます。\$HOME を使用することにより、パスは自動的にユーザの HOME 変数の値に応じて個別に設定されます。
 - <デフォルトバージョン> フィールドで、すべてのアクションを行う対象となるリビジョンを指定します。指定にはリビジョン番号またはバージョンラベルを使用します。デフォルトではチップリビジョンが使用されます。
 - <ベース>、<ブランチ>、および <デフォルト> に該当するバージョンラベルを入力し、自動ブランチを定義します。<ベースバージョン> にはブランチを開始するリビジョンを識

別するために指定する固定バージョンラベルを指定します。〈ブランチバージョン〉には、ブランチのチップに割り当てる浮動バージョンラベルを指定します。最初は、ブランチを実行するリビジョンが指定されています。以降は、ブランチリビジョンをチェックインするたびに、このラベルが自動的にブランチのチップリビジョンに割り当て直されます（浮動します）。〈デフォルトバージョン〉には、ブランチバージョンオプションに対して指定したバージョンラベルを入力します。このバージョンラベルにより、すべてのアクションの対象とするリビジョンが指定されます。259 ページの『ファイルのブランチとマージ』を参照してください。

これらの値は、これらのオプションがプロジェクトデータベースに関連付けられたマスターコンフィグレーションファイル中で変更禁止に設定されていないかぎり、プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中の値に優先します。

- 〈デフォルトプロモーショングループ〉 ドロップダウンリストから 1 つを選択して、最下位のプロモーショングループを定義します。このドロップダウンリストは、選択されたプロジェクトデータベースまたはプロジェクトにプロモーションモデルが設定されている場合にのみ使用可能となります。このオプションは、プロモーションモデルに最下位のプロモーショングループが複数存在する場合に便利です。

デフォルトプロモーショングループとは、プロモーションモデル中の最下位のプロモーショングループです。

デフォルトでは、リビジョンのチェックアウト、リビジョンのロック、および作業ファイルの追加を行うときに指定したデフォルトプロモーショングループが、リビジョンに割り当てられます。デフォルトプロモーショングループを定義しておく、ユーザがこれらのアクションを行うたびに最下位のプロモーショングループを指定する必要がなくなります。プロモーションモデルが設定されている場合は、チェックアウトおよびロックを行うために、ユーザは最下位のプロモーショングループを指定しなければなりません。

作業ファイルの追加時に、プロモーションモデルが設定されているが、最下位のプロモーショングループのデフォルト値が設定されていない場合は、その作業ファイルが、プロモーショングループが割り当てられていない新リビジョンとして追加されます。

- 9 【OK】をクリックします。これで新しいパブリックワークスペースが作成され、その設定が定義されました。プロジェクトデータベース中のプロジェクトに対してワークスペースを定義する場合は、手順 10 に進みます。
- 10 プロジェクトデータベース中のプロジェクトに対してワークスペースを定義するには、プロジェクト表示部からプロジェクトを選択し、手順 8 と 9 を繰り返します。

ワークスペースの設定

ワークスペースを設定するには

- 1 ワークスペースを設定するプロジェクトデータベース、またはプロジェクトデータベース中のプロジェクトを選択します。

- 2 [ファイル] - [ワークスペースの設定] を選択します。[ワークスペースの設定] ダイアログボックスが表示されます。



注 ルートワークスペースの左に表示される小さな青い菱形のアイコンは、このルートワークスペースがデフォルトのワークスペースとして設定されていることを示し、実行されている各 Version Manager に対してルートワークスペースが最初からプロジェクトデータベースに対して設定されていることを意味します。〈デフォルトワークスペースの作成〉チェックボックスは、この例のようにデフォルトのワークスペースがこのダイアログボックス中で選択されている場合はグレー表示になります。プロジェクトデータベースに対してほかのワークスペースを選択した場合は、このチェックボックスがアクティブになります。

- 3 プロジェクトデータベースに関連付けるワークスペースを選択します。
- 4 次回以降の Version Manager 実行時に、このワークスペースをプロジェクトデータベースに関連付ける場合は、〈デフォルトワークスペースの作成〉チェックボックスを選択します。選択しない場合は、このワークスペースは現在の Version Manager 実行時のみアクティブとなります。
- 5 【OK】をクリックします。

カスタムツールの追加

Version Manager デスクトップクライアントでは、ツールバーのアイコン、[ツール] メニューの項目、またはその両方として表示されるカスタムツールを追加できます。この機能は頻繁に使用されるアプリケーションやタスクにショートカットを設定する場合に便利です。

プロジェクトデータベースには、それぞれ独自のカスタムツールバーと [ツール] メニューを設定できます。カスタム [ツール] メニューを持つプロジェクトデータベースから持たないプロジェクトデータベースに切り換えると、その [ツール] メニューは選択不可となります。該当するツールバー

は、プロジェクトデータベースからほかのプロジェクトデータベースに移動するときに、読み込まれます。

このツール設定機能により、次のような作業を行えます。

- ユーザ定義のツールバーアイコンの追加
- ユーザ定義のツールバーアイコンの削除。デフォルトの Version Manager ツールバーアイコンは削除できません。
- メニューバーの [ツール] メニューへのメニュー項目の追加
- 各プロジェクトデータベースのツールバーおよびメニューバー設定の保存および読み込み
- ユーザ ID、アーカイブ名、および作業ファイル名などの情報の、ツールバーアイコンまたはメニュー項目に関連付けられた実行ファイルへの引き渡し

カスタムツールの追加

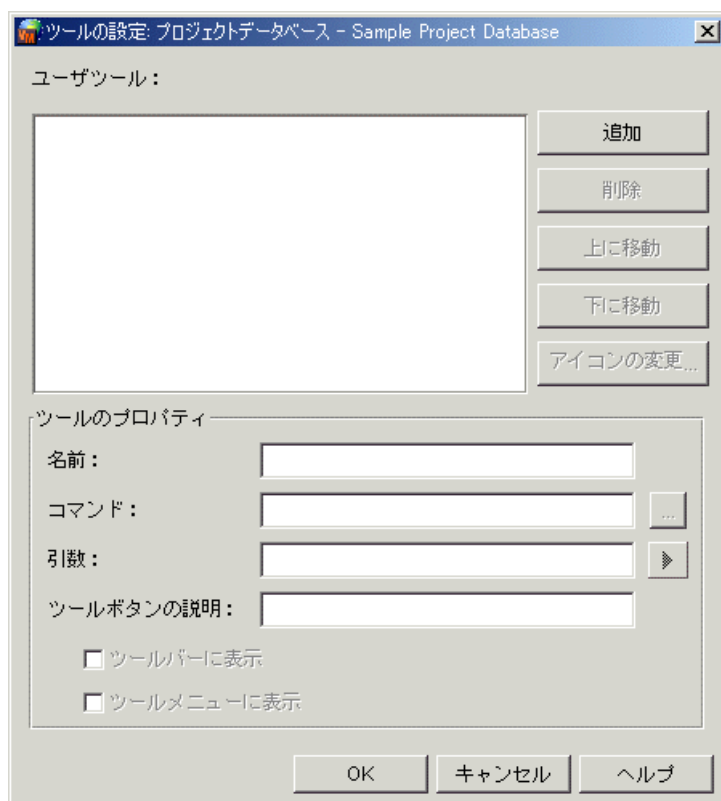
カスタムツールを追加するときには、以下を指定することができます。

- アイコンに使用する画像。Version Manager には新規ツールバーアイコンに使用するデフォルト画像 (GIF ファイル) が付属しています。独自に GIF または JPG ファイルを使用することも可能です。画像は、幅、21 ピクセル、高さ、19 ピクセルに拡大または縮小されて使用されます。
- アイコンをクリックするかメニュー項目を選択したときに実行されるコマンド (ファイル)。Version Manager では次のものがサポートされます。
 - スタンドアロンの Windows (.EXE) および UNIX 実行プログラム
 - Windows バッチスクリプト (.BAT)
 - Windows プログラム情報ファイル (.PIF)
 - UNIX シェルスクリプト
- ツールバーアイコンまたはメニュー項目に関連付けられた実行ファイルに渡すための情報 (引数)。271 ページの第 9 章『イベントトリガの使用』を参照してください。
- マウスポインタをツールバーアイコン上に置いたときに表示される、ツールのポップアップヘルプ。

カスタムツールを追加するには

- 1 プロジェクトデータベースを選択します。

- 2 [管理] - [ツールの設定] を選択します。[ツールの設定] ダイアログボックスが表示されま
す。



- 3 【追加】をクリックします。左側の〈ユーザツール〉ボックスに、デフォルトのアイコンとその
名称、「[タイトルなし]」が表示されます。手順 8 で説明されているように、このアイコンの画
像は変更できます。
- 4 〈名前〉フィールドに、新しいツールの名称を指定します。新しいツールがメニュー項目の場合
は、ここに入力するテキストが [ツール] メニューのメニュー項目名になります。新しいツール
がツールバーアイコンの場合は、ここに入力するテキストが、〈ツールボタンの説明〉フィール
ドに別のテキストを指定しないかぎり、ツールバーのポップアップヘルプになります。
- 5 〈コマンド〉フィールドから、ユーザが新しいアイコンをクリックしたとき、または新しいメ
ニュー項目を選択したときに起動される実行ファイルを指定します。
- 6 この新しいツールをツールバーアイコンにするには〈ツールバーに表示〉チェックボックスを選
択します。
- 7 この新しいツールを [ツール] メニューのメニュー項目にするには〈ツールメニューに表示〉
チェックボックスを選択します。
- 8 必要に応じて以下を指定することもできます。
 - アイコン画像の変更。【アイコンの変更】をクリックし、デフォルトの代わりに GIF ファイル
または JPG ファイルを選択します。
 - ポップアップヘルプの変更。〈名前〉フィールドに指定されたもの以外の内容に変更するに
は、〈ツールボタンの説明〉フィールドにテキストを入力します。
 - 〈コマンド〉で指定された実行ファイルへの情報 (引数) の転送。〈引数〉フィールドの隣の
ボタンをクリックし、実行ファイルにイベント情報を送るコマンドラインマクロまたはパラ
メータファイル (__EventParmFile__) を選択します。



アイコンまたはメニュー項目イベントトリガへの情報転送方法については、276 ページの『[イベントトリガへのイベント情報の転送](#)』を参照してください。

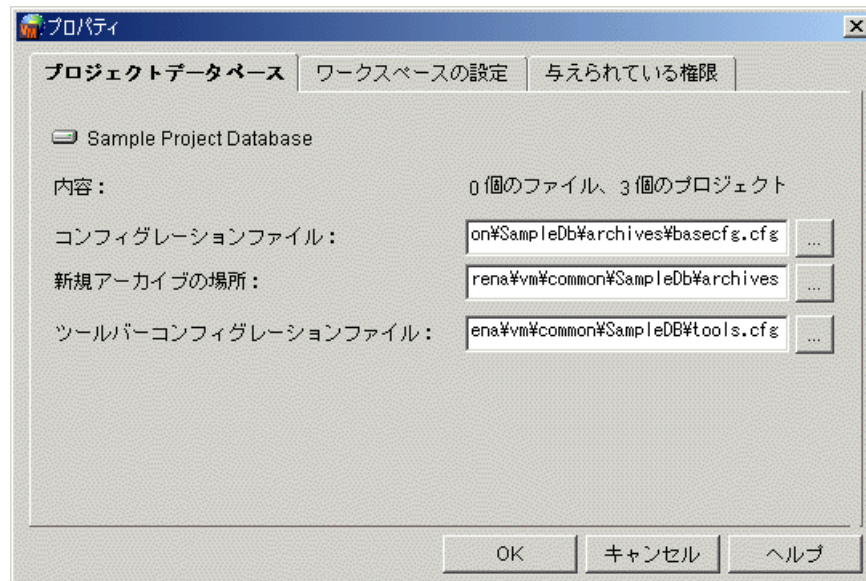
- 9 【OK】をクリックします。デフォルトではツール設定情報はプロジェクトデータベースの場所の下の tools.cfg というファイルに保存されます。このファイルは、新しい場所を指定しないかぎり移動しないでください。移動すると、ツールバーとメニューバーをカスタマイズできなくなります。

ツール設定ファイルの移動

ツール設定ファイル (tools.cfg) は新しい場所に移動することができます。移動する場合は、ツールバーとメニューバーをカスタマイズできるよう、新しい場所を指定する必要があります。ツール設定ファイルは 5.3 プロジェクトデータベースには移動できません。

ツール設定ファイルの新しい場所を指定するには

- 1 ツール設定ファイルに関連付けられたプロジェクトデータベースを選択します。
- 2 [ファイル] - [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示されます。



- 3 〈コンフィグレーションファイル〉テキストボックスに移動先の場所を入力します。または、テキストボックスの後の [...] をクリックして、移動先の場所を選択します。
- 4 【OK】をクリックします。

既存アーカイブの属性の変更

アーカイブの属性がコンフィグレーションファイル内の設定に基づいて設定されるか、Version Manager がコンフィグレーションファイルを使用するように設定されていない場合は、アーカイブの作成時に Version Manager のデフォルトに基づいてアーカイブの属性が設定されます。Version Manager のデフォルト値のリストは、74 ページの『[コンフィグレーションファイルが使用されない場合のデフォルト設定](#)』を参照してください。

アーカイブが作成された後に、既存のアーカイブのいくつかの設定を変更できます。既存のアーカイブの属性を変更した場合は、そのアーカイブのすべてのリビジョンで属性が変更されます。

属性の変更が必要な場合

既存のアーカイブの属性は、新しく作成されるアーカイブの属性を決定する設定を変更する場合のみ変更してください。たとえば、複数のロックを許可しない属性を持つアーカイブがあり、複数のロックを許可するように設定を変更した場合は、複数のロックが新しく作成されるアーカイブでのみ認められます。次にこのような状況が発生するシナリオを示します。

小規模な開発グループのプロジェクトリーダーが、新しい Windows 版製品の最初のバージョンを発表したとします。この製品がリリースされるまで、開発者はトランクチップのリビジョンにのみ作業が必要でした。プロジェクトリーダーがアーカイブを作成する前に、開発者が不注意にブランチを作成しないよう、このリーダーはアーカイブごとに 1 つだけロックを許可するようにアーカイブを設定しました。

次に、この製品の UNIX 版を開発しながら、Windows 版の次バージョンを開発するとします。このため、各チップリビジョンごとに、UNIX 版のためにブランチを作成することにします。並行開発を行うには、ユーザがアーカイブ内の複数のリビジョンに対してロックを設定する必要があるため、既存のプロジェクトファイルの属性を変更する必要があります。

アーカイブ属性

変更可能なアーカイブ属性を次に示します。

属性	定義
排他ロック	単一のアーカイブに対する複数のロックを防止します。
キーワード展開	作業ファイルのチェックイン時にキーワードの展開を可能にします。
コメント接頭辞	\$Log\$ キーワード展開時に行の前に挿入するコメント接頭辞を定義します。
書き込み禁止	不注意による削除や変更からアーカイブを保護します。
デルタ形式で格納	チップリビジョンのみを作業ファイルの完全なコピーとして保存し、その他のリビジョンはすべてデルタとして保存します。
改行コード	\$Log\$ キーワード展開時の改行コードを定義します。
改行コードの変換	EOL シーケンスを変換します。
所有者	アーカイブの所有者を指定します。
アクセスリスト	アーカイブのアクセスリストを指定します。
レコード長	固定長と可変長のどちらのファイルを保存するかを指定します。固定長の場合には各レコードの長さを指定する必要があります。
カラムマスク/リナンバ	行番号を含むファイルに対するオプションを指定します。

ファイルタイプによって設定される属性

前述の属性のいくつかはファイルタイプに基づいて設定され、ほかのものはアーカイブに保存されるすべてのファイルタイプに基づいて設定されます。たとえば、Translate EOL 属性はテキストファイルに対してのみ設定されますから、ファイルタイプに基づいて設定されます。これは、バイナリファイルに設定するとファイルが破損するためです。排他ロックは通常、ファイルタイプにかかわらず、

すべてのアーカイブに対して同様に設定されます。ファイルタイプに基づいて設定される属性は次のとおりです。

■ キーワード展開	■ 改行コード
■ デルタ形式で格納	■ レコード長
■ 改行コードの変換	■ リナンバ
■ カラムマスク	

既存の属性の判断

既存のアーカイブの属性を判断するには、アーカイブ情報のみが含まれる履歴レポートを生成し、レポートの「属性」セクションを見ます。[297 ページの『履歴レポートの生成』](#)を参照してください。

デスクトップクライアントを使用する場合

デスクトップクライアントでは、プロジェクトデータベース、プロジェクト、複数のバージョンを持つファイル、または単一バージョンのファイルに関連付けられた既存アーカイブの属性を変更できません。プロジェクトデータベースまたはプロジェクトを選択して、既存アーカイブの属性を次のように変更することができます。

- 選択されたプロジェクトデータベースまたはプロジェクト内の、すべてのプロジェクト / サブプロジェクト
- プロジェクトまたはサブプロジェクトを含まない、選択されたプロジェクトデータベースまたはプロジェクトのみ

ファイルタイプに基づいて設定される属性を変更する（[186 ページの『ファイルタイプによって設定される属性』](#)参照）場合は、フィルタ（[表示] - [フィルタ] - [ワイルドカード]）を使用して、特定の条件を満たすすべてのバージョンのファイルを表示できます。たとえば、拡張子 .java を持つファイルをすべて表示し、そこからバージョン管理ファイルを選択できます。

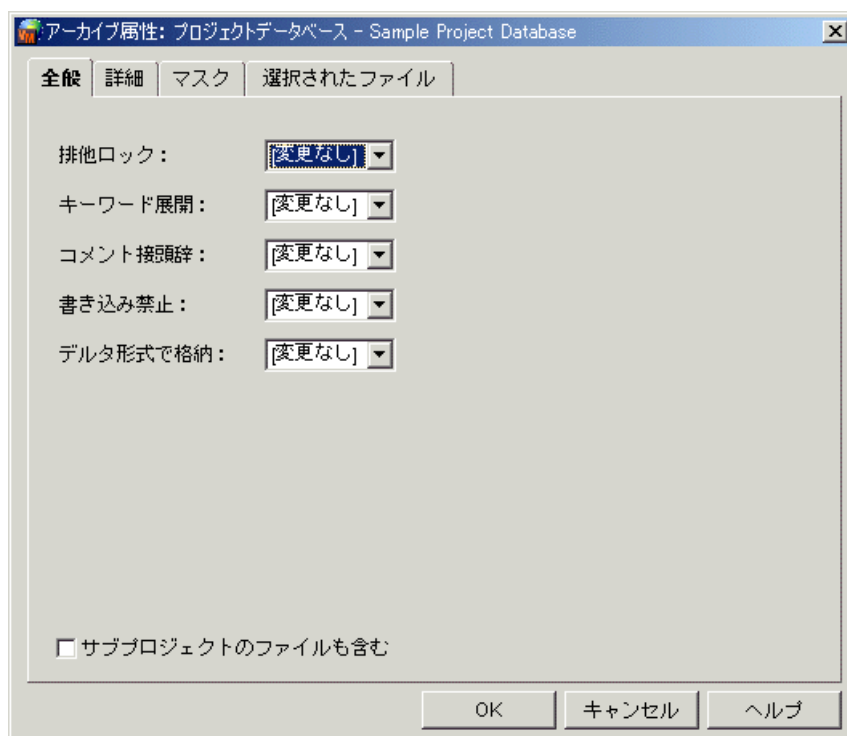


注 ユーザは、アーカイブ属性の変更時に、複数のプロジェクトおよびフォルダを選択できます。複数のプロジェクトおよびフォルダの選択については、『Serena ChangeMan Version Manager ユーザガイド』を参照してください。

アーカイブ属性を変更するには

- 1 プロジェクトデータベース、プロジェクト、またはバージョン管理ファイルを選択します。プロジェクトデータベースまたはプロジェクトを選択し、ファイルタイプを基にして設定される属性を変更する場合は、その属性はプロジェクトデータベースまたはプロジェクト中のすべてのファイルタイプについて変更されます。

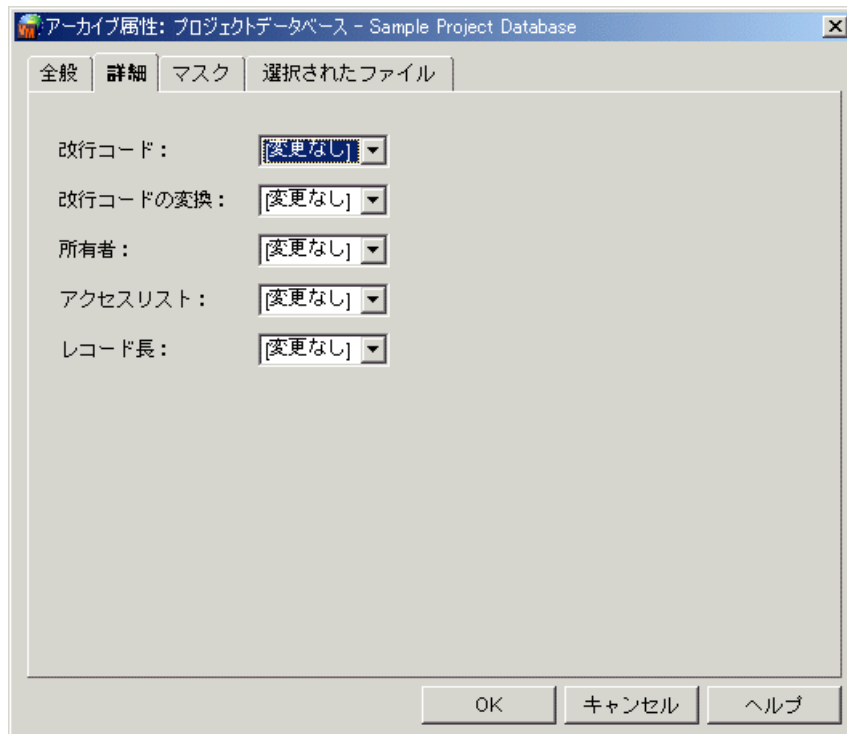
- 2 [管理] - [アーカイブ属性] を選択します。[アーカイブ属性] ダイアログボックスが表示され、[全般] タブがアクティブになっています。



- 3 [全般] タブの各属性の設定は「変更しない」になっており、属性には現在の設定が使用されます。設定を変更するには、次の作業を行います。
- <排他ロック> で「はい」または「いいえ」を選択します。「はい」を選択すると、単一アーカイブへの複数のロックが防止されます。「いいえ」を選択して複数のロックを許可すると、アーカイブ中の複数のリビジョンを同時にロックすることが可能となります。さらに、アーカイブに対して、ここで「はい」を指定すると、そのアーカイブでは MultiLock ディレクティブが無視されます。84 ページの『アーカイブ作成時のオプション』を参照してください。
 - <キーワード展開> で「はい」または「いいえ」を選択します。「はい」を選択すると、作業ファイルのチェックイン時にキーワードが展開されます。この属性はファイルタイプに基づいて設定されます。112 ページの『ファイルタイプに基づいて設定されるオプション』を参照してください。このオプションは、バイナリファイルを破壊する可能性があるため、バイナリファイルには設定しないでください。
 - <コメントの接頭辞> フィールドは、キーワードを展開する場合にのみ適用されます。この場合、このフィールドは \$Log\$ キーワード展開時の行の前に挿入するコメント接頭辞を定義します。<コメントの接頭辞> では、「変更」または「削除」を選択します。「変更」を選択すると、コメント接頭辞を入力するための <コメントの接頭辞> テキストフィールドが表示されます。「削除」を選択すると、以前に指定されたコメント接頭辞が削除されます。この属性はファイルタイプに基づいて設定されます。112 ページの『ファイルタイプに基づいて設定されるオプション』を参照してください。
 - <書き込み禁止する> で「はい」または「いいえ」を選択します。「はい」を選択すると、意図しない削除または変更からアーカイブが保護されます。Version Manager のコマンドは、アーカイブの変更前に自動的に書き込み保護を解除し、終了後に元に戻します。
 - <デルタ形式で格納> で「はい」または「いいえ」を選択します。作業ファイルの完全なコピー（作業イメージ）はチップリビジョンに対してのみ保存され、ほかのすべてのリビジョ

ンはデルタとして保存されます。つまり、すべてのリビジョンが完全な作業ファイルのコピーとしては保存されません。この属性はファイルタイプに基づいて設定されます。バイナリファイルについては、リビジョン間の差が大きい場合があるため、このオプションを「いいえ」に設定することをお勧めします。113 ページの『デルタの保存』を参照してください。

- 4 手順 1 でプロジェクトデータベースまたはプロジェクトを選択している場合は、すべてのプロジェクト / サブプロジェクトの既存のアーカイブの属性を変更するには〈サブプロジェクトのファイルも含む〉チェックボックスを選択します。
- 5 [詳細] タブを選択します。



- 6 [詳細] タブ上の各属性の設定は「変更しない」になっており、属性には現在の設定が使用されます。設定を変更するには、次の作業を行います。
 - 〈改行コード〉フィールドは、キーワードを展開する場合にのみ適用されます。この場合、このフィールドは \$Log\$ キーワード展開の改行コードを定義します。ここで、「変更」または「削除」を選択します。「変更」を選択すると、改行コードを入力するための〈改行コード〉テキストフィールドが表示されます。「削除」を選択すると、以前に指定された改行コードが削除されます。この属性はファイルタイプに基づいて設定されます。112 ページの『ファイルタイプに基づいて設定されるオプション』を参照してください。



注 Windows および UNIX プラットフォーム間の改行コードの変換は、自動的に処理されません。

- 〈改行コードの変換〉で「はい」または「いいえ」を選択。このオプションはテキストファイルについては常に「はい」に設定します。この属性はファイルタイプに基づいて設定されま

す。112 ページの『ファイルタイプに基づいて設定されるオプション』を参照してください。



注意！ このオプションはバイナリファイルには設定しないでください。アーカイブを破損する場合があります。

- 〈所有者〉で「変更」または「削除」を選択します。「変更」を選択すると、は新しいアーカイブの所有者のユーザ ID を入力する〈所有者〉が表示されます。「削除」を選択すると、以前に指定された所有者がアーカイブから削除されます。84 ページの『アーカイブ作成時のオプション』を参照してください。
 - 〈アクセスリスト〉で「変更」、「追加」、または「削除」を選択します。〈変更〉または「追加」を選択すると、〈アクセスリスト〉フィールドが表示されます。「変更」の場合は、このフィールドに新しいアクセスリストを入力します。「追加」の場合は、既存のアクセスリストにユーザ ID とグループを追加入力します。このフィールドの値は、合計で 254 文字を超えることはできません。「削除」を選択すると、既存のアクセスリストが削除されます。アクセスリストについては、201 ページの『セキュリティの使用』を参照してください。
 - 〈レコード長〉で「可変長」または「固定長」を選択します。「可変長」を選択すると、アーカイブに可変長のデータが保存されるよう Version Manager が設定されます。「固定長」を選択すると、アーカイブに固定長のデータが保存されるよう Version Manager が設定されます。「固定長」が選択されると、レコードの長さを入力するための〈レコード長〉テキストフィールドが表示され、より効率的にデルタが生成されます。この属性はファイルタイプに基づいて設定されます。112 ページの『ファイルタイプに基づいて設定されるオプション』を参照してください。
- 7 行番号を含むファイルの作成と更新にのみ適用される属性を修正するには、[マスク] タブを選択します。[マスク] タブの各属性の設定は「変更しない」に設定されており、この場合は現在の属性の設定が使用されます。

アーカイブ属性: プロジェクトデータベース - Sample Project Database

全般 詳細 **マスク** 選択されたファイル

マスク

- 変更なし
- マスクの削除
- COBOL マスクを使用
- マスクするカラムの範囲: [] (*は数値の範囲を示します)

リナンバ

- 変更なし
- リナンバの削除
- COBOL リナンバを使用
- リナンバ:

リナンバの開始: [] リナンバの増加分: []

開始カラム: [] 終了カラム: []

OK キャンセル ヘルプ

- 8 [マスク] タブで、次の属性を変更します。これらの属性はファイルタイプに基づいて設定されます。
- <マスク> グループから、次のいずれかのオプションを選択します。
 - <マスクの削除> を選択すると、マスクは使用されません。
 - <COBOL マスクを使用> を選択すると、COBOL ファイルに対し、デフォルトの COBOL マスクが使用されます。この場合、第 73-80 桁がマスクされ、第 1 桁目が数値の場合には第 1-6 桁もマスクされます。
 - <マスクするカラムの範囲> には、マスクを設定する最初と最後の桁を入力します。たとえば、1-6 と入力すると、第 1 桁から第 6 桁までがスペースに変換されます。
カラムマスクを数値フィールドのみに限定するには、範囲の次にアスタリスク (*) を入力します。たとえば、1-6* では、第 1 桁目が数値の場合にのみマスクが適用されます。
また、複数のカラムの範囲をコンマで区切って入力できます。たとえば、1-6*,10-12 と入力すると、第 1 桁が数値である場合にのみマスクが適用され、第 10 桁から第 12 桁までがスペースに変換されます。
 - <リナンバ> グループから、次のいずれかのオプションを選択します。
 - <リナンバの削除> を選択すると、リナンバは行われません。
 - <COBOL リナンバを使用> を選択すると、COBOL ファイルに対してデフォルトの COBOL リナンバの定義が使用されます。この場合、第 1 桁から第 6 桁目までを、10 から開始して 10 ずつ増加する番号で再設定します。
 - <リナンバ> には、以下のように入力します。
 - a <リナンバの開始> フィールドには、最初の行番号を入力します。
 - b <リナンバの増加分> フィールドには、行番号の増加分を入力します。たとえば、最初の番号に 20 を入力し、増加分を 5 と入力した場合は、最初の行の番号が 000020、次の行が 000025、3 番めが 000030、のようになります。ここでは 6 桁にリナンバするものと仮定しています。
 - c <開始カラム> フィールドには、リナンバ元となる桁、<終了カラム> フィールドは、リナンバ先となる桁を入力します。
- 9 【OK】 をクリックします。

コマンドラインインターフェイスを使用する場合

コマンドラインインターフェイスでは、VCS コマンドを使用してアーカイブ属性を変更します。次の表に、このコマンドに対するアーカイブ属性オプションのリストを示します。

変更する属性	使用するコマンド
排他ロック	vcs [- +]pe
キーワード展開	vcs [- +]pk
コメント接頭辞	vcs -ecstring
書き込み禁止	vcs [- +]pw
デルタ形式で格納	vcs [- +]pg
改行コード	vcs -enstring
改行コードの変換	vcs [- +]pt

変更する属性	使用するコマンド
所有者	<code>vcs -ouser_id</code>
アクセスリスト	<code>vcs -a[user_id group[,user_id group...]]</code>
レコード長	<code>vcs -xrecordlength=record_length</code>
カラムマスク	<code>vcs -xcolumnmask="start-end[(numeric)]..."</code>
リナンバ	<code>vcs -xrenumber=start-end [from start by number]</code>

VCS コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

アーカイブの移動

デフォルトでは、プロジェクトデータベースまたはプロジェクトを作成すると、アーカイブがプロジェクトデータベースの下の場所に作成されます。データベースまたはプロジェクトの作成時には、ほかの場所も指定できます。プロジェクトデータベース作成時に指定したアーカイブの場所が不適切になった場合は、その場所を変更できます。アーカイブの場所にアーカイブファイルを保存する前に、場所を変更したほうが作業の効率が上がります。

アーカイブファイルを格納した後で移動しなければならなくなった場合は、バージョン管理ファイルとアーカイブの間の参照を回復するために、新しい場所からアーカイブをインポートする必要があります。

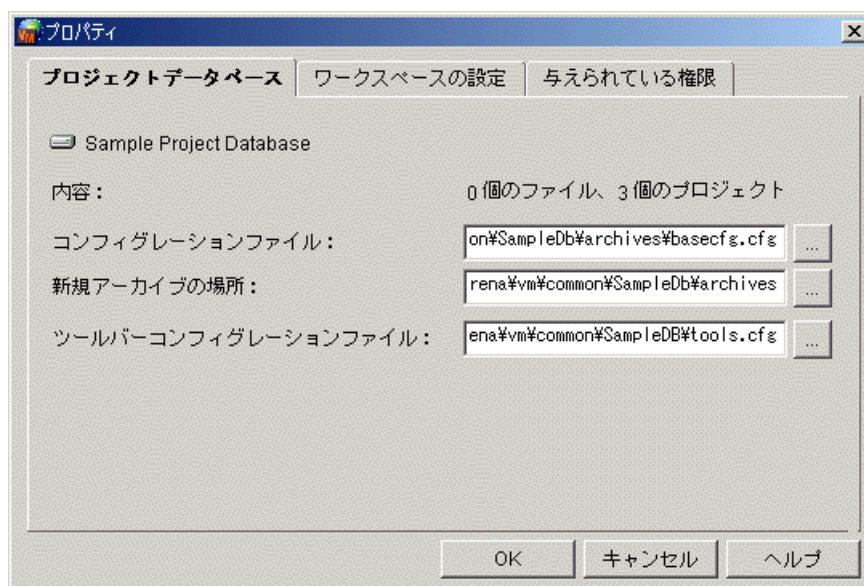
アーカイブの場所の変更

この手順では、プロジェクトデータベースが作成され、アーカイブの場所が指定されてから、その場所にアーカイブファイルを保存する前（アーカイブの場所にはまだ、アーカイブファイルが存在しないとき）にアーカイブの場所を変更すると仮定しています。

アーカイブの場所を変更するには

- 1 プロジェクトデータベースまたはプロジェクトを選択します。

- 2 [ファイル] - [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示されます。



- 3 〈新規アーカイブの場所〉フィールドに新しい場所のパスを入力します。
- 4 [OK] をクリックします。

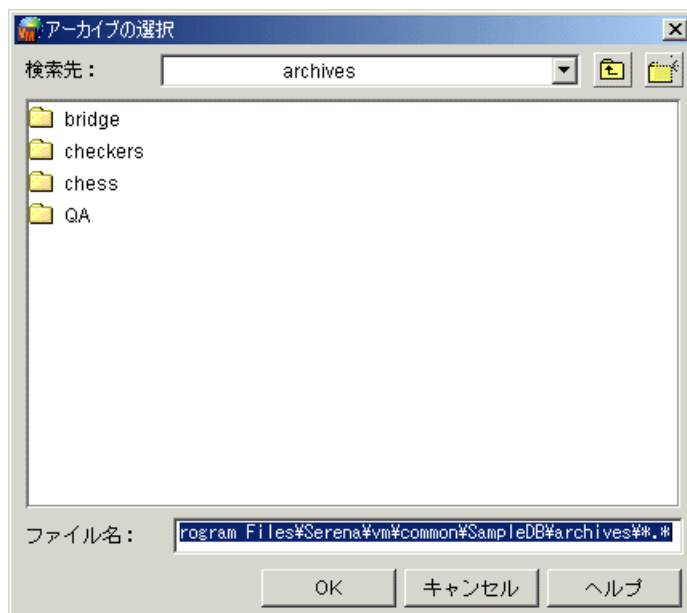
アーカイブのインポート

この手順では、プロジェクトのアーカイブを新しい場所に移動したと仮定します。アーカイブをインポートする前に、プロジェクトから既存のバージョン管理ファイルを削除する必要があります。名称の衝突を避けるため、プロジェクト中に同じ名称のバージョン管理ファイルが存在する場合は、アーカイブをインポートすることはできません。新しい場所からアーカイブをインポートすると、アーカイブのバージョン管理ファイルがプロジェクトに含められ、バージョン管理ファイルとアーカイブの間の参照が回復されます。

アーカイブをインポートするには

- 1 アーカイブを移動したバージョン管理ファイルを選択します。すべて選択するには [編集] - [すべて選択] を選択します。
- 2 [ファイル] - [削除] を選択します。
- 3 アーカイブのインポート先のプロジェクトを選択します。

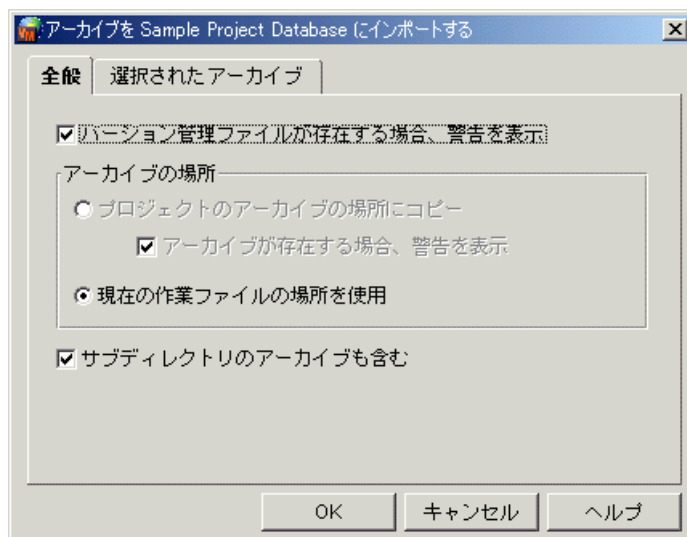
- 4 [管理] – [アーカイブのインポート] を選択します。[アーカイブの選択] ダイアログボックスが表示されます。このダイアログボックスには、選択されたプロジェクトのアーカイブの場所が表示されます。



- 5 アーカイブの新しい場所（アーカイブの移動先の場所）を表示します。次にアーカイブを含むフォルダ全体を選択するか、インポートするアーカイブのみを選択し、【OK】をクリックします。[アーカイブのインポート] ダイアログボックスが表示され、[全般] タブがアクティブになっています。



注 〈ファイル名〉フィールドで、* または *.* 以外のフィルタを持つパス（例：c:¥files¥*.cpp を指定すると、フィルタに一致するファイルは最上レベルのプロジェクトを作成せずに直接追加されます。ただし、サブディレクトリも追加される場合は、サブプロジェクトがそれらに対して作成され、フィルタに一致するファイルのみが追加されます。



- 6 〈アーカイブの場所〉グループから〈既存のアーカイブディレクトリを使用〉を選択します。これによりプロジェクトのアーカイブの場所が新しい場所に変更されます。

- 7 フォルダ全体を選択している場合は、〈サブプロジェクトのアーカイブも含む〉チェックボックスを選択して、アーカイブ構造に含まれるすべてのサブディレクトリ中のすべてのアーカイブを追加できます。
- 8 選択したアーカイブをインポートする前に確認するには、〔選択されたアーカイブ〕タブをクリックします。
- 9 【OK】をクリックします。

プロジェクトとプロジェクトデータベースのコピー

プロジェクトとプロジェクトデータベースのコピーを作成できます。これは、新規プロジェクトデータベースの作成または新しいシステムへの移行のショートカットとして行うことができます。



注 既存の非ファイルサーバプロジェクトデータベースを Version Manager ファイルサーバへ移行する方法の詳細は、149 ページの『Version Manager File Server へのプロジェクトデータベースの追加』を参照してください。

プロジェクトのコピー

プロジェクトのコピーでは、以下を行うことができます。

- 全サブプロジェクトをコピーするか、またはプロジェクトのみをコピーする。
- アーカイブを新しい場所にコピーするか、または既存のアーカイブの場所を使用する。
- 新しいコンフィグレーションファイルを作成するか、既存のコンフィグレーションファイルを新しいプロジェクトの場所にコピーするか、または既存のコンフィグレーションファイルを使用する。
- 新規アクセスコントロールデータベースを作成するか、既存のアクセスコントロールデータベースを新しいプロジェクトの場所にコピーするか、または既存のアクセスコントロールデータベースを使用する。



注 プロジェクトをコピーすると、プロジェクトのワークスペース設定は保持されません。

システムデフォルトの使用

デフォルトでは、ファイルをコピーすると、次の動作が行われます。

- 既存のアーカイブの場所でもとのプロジェクトのアーカイブを参照する。コピーされたサブプロジェクトもその既存のアーカイブの場所を参照します。複数のサブプロジェクトを含むプロジェクトをコピーする場合は、サブプロジェクトのアーカイブの場所が異なる場合があることに注意してください。プロジェクトをコピーして、既存のアーカイブの場所を使用する場合、新規プロジェクトはそれぞれのアーカイブの場所すべてを参照します。



注 コピーされた後にプロジェクトに追加された新規アーカイブは、新しいプロジェクトのアーカイブの場所を使用します。デフォルトは、プロジェクトデータベースの場所にあるアーカイブディレクトリの下プロジェクトの名称のディレクトリです。

- プロジェクトの既存のコンフィグレーションファイルを使用する。

- プロジェクトの既存のアクセスコントロールデータベースを使用する。
- サブプロジェクトをコピープロジェクト操作対象に追加する。
- 作業ファイルの場所として、プロジェクトデータベースの場所の下にある work という名称のディレクトリを使用する。プロジェクトからバージョン管理ファイルをチェックアウトすると、Version Manager は作業ディレクトリの下プロジェクトの名称が付いたディレクトリを作成し、それを新しいプロジェクトのデフォルトの作業ファイルの場所として使用します。(プロジェクトをコピーした後デフォルトの作業ファイルの場所を変更するには、[ファイル] - [プロパティ] を選択し、[ワークスペースの設定] タブを選択し、〈作業ファイルの場所〉フィールドを修正します。)

プロジェクトをコピーする際、これらのデフォルトはどれでも変更することができます。以下にその手順を説明します。

プロジェクトをコピーするには

- 1 プロジェクト表示部で、コピーするプロジェクトを選択します。
- 2 [編集] - [コピー] を選択します。[コピープロジェクト] ダイアログボックスが表示されます。
- 3 コピー先のプロジェクトデータベースまたはプロジェクトを選択し、【次へ】をクリックします。プロジェクトを開いているプロジェクトデータベースまたはプロジェクトにコピーすることができます。2 番目の [コピープロジェクト] ダイアログボックスが表示されます。
- 4 【次へ】をクリックしてデフォルトをそのまま使用するか、以下のフィールドを編集してデフォルトを変更します。

操作	手順
プロジェクトのアーカイブディレクトリ (とその内容) をコピーしてコピー先プロジェクトデータベースのアーカイブディレクトリの下に置く。 (新しいアーカイブディレクトリの場所は、アーカイブディレクトリ内のコピー先プロジェクトデータベースの階層構造と同じになります。)	〈プロジェクトの場所にアーカイブをコピー〉オプションを選択します。
新しいアーカイブのベースリビジョン (開始点) として指定リビジョンの選択を許可する。	〈指定したリビジョンを保持し、履歴を破棄〉チェックボックスを選択します。 このフィールドは、〈プロジェクトの場所にアーカイブをコピー〉オプションが選択されている場合にのみ使用できることに注意してください。
Version Manager が初期リビジョン (またはベースライン) として使用するリビジョンを選択する。	初期リビジョンを選択するには、〈デフォルトリビジョン〉フィールドに、バージョンラベル、またはプロモーショングループを入力するか、【...】をクリックして選択します。
サブプロジェクトとそのファイルを含める。	〈サブプロジェクトを含む〉チェックボックスを選択します。

- 5 3 番目の [コピープロジェクト] ダイアログボックスで、デフォルトをそのまま使用して【終了】をクリックするか、以下のフィールドを編集してデフォルトを変更します。

操作	手順
ファイルを新しいプロジェクトの場所へコピーせずに既存のコンフィグレーションファイルを使用する。	〈既存のコンフィグレーションファイルを使用〉オプションを選択します。コピーされたプロジェクトは、もとのプロジェクトとコンフィグレーションファイルを共有します。
既存のコンフィグレーションを新しいプロジェクトへコピーする。	すべての既存コンフィグレーションファイルを新しいプロジェクトの場所へコピーするには、〈既存のコンフィグレーションファイルをコピー〉を選択します。
新しいプロジェクトの新しいデフォルトコンフィグレーションファイルを作成する。	〈新規コンフィグレーションファイルを作成〉オプションを選択します。Version Manager は、default.cfg ファイルをコピーしてデフォルトのコンフィグレーションファイルを作成し、プロジェクトデータベースの場所の下のアーカイブディレクトリに入れます。
すべての既存のアクセスコントロールデータベースを新しいプロジェクトへコピーする。	〈アクセスコントロールを他の場所にコピー〉オプションを選択します。プロジェクトのアクセスコントロールデータベースは、プロジェクトデータベースの場所の下のアーカイブディレクトリに入れられます。プロジェクトアクセスコントロールデータベース（存在する場合は、アーカイブディレクトリの下プロジェクトの名称の付いたディレクトリに入れられます。 注 このオプションは、既存のコンフィグレーションファイルを使用する場合には使用できません。
新規プロジェクトの新しいアクセスコントロールデータベースを作成する。	〈新規アクセスコントロールデータベースを作成〉オプションを選択します。このアクセスコントロールデータベースは、default.db ファイルをコピーして作成されます。この新しいアクセスコントロールデータベースはコンフィグレーションファイルでは無効になっています。デフォルトでは、新しいアクセスコントロールデータベースはアーカイブディレクトリにあります。 注 このオプションは、既存のコンフィグレーションファイルを使用する場合には使用できません。

- 6 【終了】をクリックします。

プロジェクトデータベースのコピー

Version Manager プロジェクトデータベースのコピーにより、プラットフォームデータベースが新しいプロジェクトデータベースにコピーされます。プロジェクトデータベースのコピーでは、以下を行うことができます。

- 全プロジェクトとサブプロジェクトをコピーするか、またはプロジェクトのみをコピーする。
- アーカイブを新しい場所にコピーするか、または既存のアーカイブの場所を使用する。
- 新しいコンフィグレーションファイルを作成するか、既存のコンフィグレーションファイルを新しいプロジェクトデータベースへコピーするか、または既存のコンフィグレーションファイルを使用する。
- 新規アクセスコントロールデータベースを作成するか、アクセスコントロールデータベースを新しいプロジェクトデータベースへコピーするか、または既存のアクセスコントロールデータベースを使用する。
- 共有アーカイブとの関連を保持する。
- ワークスペースをコピーする。
- ツールバーコンフィグレーションファイルを新しいプロジェクトデータベースの場所へコピーする。

システムデフォルトの使用

デフォルトでは、プロジェクトデータベースをコピーすると、次の動作が行われます。

- 既存のアーカイブの場所でもとのプロジェクトのデータベースアーカイブを参照する。コピーされたサブプロジェクトもその既存のアーカイブの場所を参照します。複数のサブプロジェクトを含むプロジェクトデータベースをコピーする場合は、サブプロジェクトのアーカイブの場所が異なる場合があることに注意してください。プロジェクトデータベースをコピーして、既存のアーカイブの場所を使用する場合、新規プロジェクトデータベースはそれぞれのアーカイブの場所すべてを参照します。



注 コピーされた後にプロジェクトデータベースに追加された新規アーカイブは、プロジェクトデータベースのアーカイブの場所を使用します。デフォルトでは、プロジェクトデータベースの場所のアーカイブディレクトリです。

- プロジェクトデータベースの既存のコンフィグレーションファイルを使用する。コピーされたサブプロジェクトは、既存のコンフィグレーションファイル（存在する場合）も使用します。
- プロジェクトデータベースの既存のアクセスコントロールデータベースを使用する。コピーされたサブプロジェクトは、既存のアクセスコントロールデータベース（存在する場合）も使用します。
- ワークスペースをコピーする。

これらのデフォルトは、プロジェクトデータベースをコピーするときに次の手順で変更することができます。

プロジェクトデータベースをコピーするには

- 1 プロジェクト表示部で、コピーするプロジェクトデータベースを選択します。
- 2 [編集] - [コピー] を選択します。[コピープロジェクトデータベース] ダイアログボックスが表示されます。
- 3 <名前> フィールドにプロジェクトデータベースの名前を入力します。名前の前後にタブや空白を付けることはできません。名前にはすべての文字を使用できます。

- 4 <場所> フィールドにプロジェクトデータベースの場所を入力するか、【...】をクリックして場所を選択します。この場所は、プロジェクトデータベースの作業を行うすべてのユーザがアクセスできる必要があります。
- 5 アーカイブのデフォルトの場所を変更する場合は、<アーカイブの場所> フィールドに場所を入力するか、または【...】をクリックして場所を選択してください。デフォルトのアーカイブの場所は、プロジェクトデータベースの場所の下の archives という名称のディレクトリです。プロジェクトデータベースの新しいアーカイブがこのディレクトリに格納されます。



注 次の画面で <プロジェクトの場所にアーカイブをコピー> を選択しなければ、プロジェクトデータベースの既存のアーカイブはこのディレクトリにコピーされません。

- 6 <作業ファイルの場所> フィールドにプロジェクトデータベースの作業ファイルの場所を入力するか、【...】をクリックして場所を選択します。ほとんどのユーザにアクセス可能な、ネットワーク上のドライブを指定することを推奨します。ユーザは必要に応じて、作業ファイルの場所を変更するためのプライベートなワークスペースをローカルドライブに作成することができます。
- 7 【次へ】をクリックします。2 番目の [コピープロジェクトデータベース] ダイアログボックスが表示されます。
- 8 【次へ】をクリックしてデフォルトをそのまま使用するか、以下のフィールドを編集してデフォルトを変更します。

操作	手順
アーカイブをプロジェクトデータベースの場所（または指定したディレクトリ）の下のアーカイブディレクトリにコピーする。	<プロジェクトの場所にアーカイブをコピー> オプションを選択します。
新しいアーカイブのベースリビジョン（開始点）として指定リビジョンの選択を許可する。	<指定したリビジョンを保持し、履歴を破棄> チェックボックスを選択します。 注 このフィールドは、<プロジェクトの場所にアーカイブをコピー> オプションが選択されている場合にのみ使用できません。
Version Manager が初期リビジョン（またはベースライン）として使用するリビジョンを選択する。	初期リビジョンを選択するには、<デフォルトリビジョン> フィールドに、バージョンラベル、またはプロモーショングループを入力するか、【...】をクリックして選択します。
共有アーカイブとの関連を保持する（アーカイブの新しい個別コピーを作成せず）。	<共有アーカイブの保持> オプションを選択します。 注 このフィールドは、<プロジェクトの場所にアーカイブをコピー> オプションが選択されている場合にのみ使用できません。
サブプロジェクトとそのファイルを含める。	<サブプロジェクトを含む> チェックボックスを選択します。

- 9 3番目の〔コピープロジェクトデータベース〕ダイアログボックスで、デフォルトをそのまま使用して【終了】をクリックするか、以下のフィールドを編集してデフォルトを変更します。

操作	手順
ファイルを新しいプロジェクトデータベースへコピーせずに既存のコンフィグレーションファイルを使用する。	〈既存のコンフィグレーションファイルを使用〉オプションを選択します。コピーされたプロジェクトデータベースは、もとのプロジェクトデータベースとコンフィグレーションファイルを共有します。
既存のコンフィグレーションを新しいプロジェクトデータベースへコピーする。	〈既存のコンフィグレーションファイルをコピー〉オプションを選択します。
新しいプロジェクトデータベースの新しいデフォルトコンフィグレーションファイルを作成する。	〈新規コンフィグレーションファイルを作成〉オプションを選択します。Version Manager は、default.cfg ファイルをコピーしてデフォルトのコンフィグレーションファイルを作成し、コピーするプロジェクトデータベースの下のアーカイブディレクトリに入れます。
すべての既存のアクセスコントロールデータベースを新しいプロジェクトデータベースへコピーする。	〈アクセスコントロールを他の場所にコピー〉オプションを選択します。プロジェクトデータベースのアクセスコントロールデータベースは、コピーするプロジェクトデータベースの下のアーカイブディレクトリに入れられます。プロジェクトアクセスコントロールデータベース（存在する場合は、アーカイブディレクトリの下のプロジェクトの名称の付いたディレクトリに入れられます。 注 このオプションは、既存のコンフィグレーションファイルを使用する場合には使用できません。
新規プロジェクトデータベースの新しいアクセスコントロールデータベースを作成する。	〈新規アクセスコントロールデータベースを作成〉オプションを選択します。このアクセスコントロールデータベースは、default.db ファイルをコピーして作成されます。この新しいアクセスコントロールデータベースはコンフィグレーションファイルでは無効になっています。デフォルトでは、新しいアクセスコントロールデータベースはアーカイブディレクトリにあります。 注 このオプションは、既存のコンフィグレーションファイルを使用する場合には使用できません。
既存のワークスペース定義を新しいプロジェクトデータベースへコピーする。	〈ワークスペースをコピー〉オプションを選択します。

- 10 【終了】をクリックします。

第 6 章

セキュリティの使用

はじめに	202
アクセスコントロールデータベースについて	202
アクセスリストについて	205
権限について	206
セキュリティの計画	208
セキュリティの設定	209
Version Manager へのアクセスコントロールデータベースの埋め込み	227
プロジェクトデータベース作成の制限	228
セキュリティのメンテナンス	229
権限の定義	238

はじめに

Serena ChangeMan Version Manager では、プロジェクトとアーカイブにアクセスできるユーザおよび許可されたユーザが、そのプロジェクトとアーカイブに対して実行できるアクションを制御することができます。Version Manager のセキュリティは、次の機能を使用して定義されます。

- アクセスコントロールデータベース
- アクセスリスト
- 権限

アクセスコントロールデータベースには、プロジェクト、すなわちそのプロジェクトのアーカイブにアクセスできるかが定義されます。アクセスリストはさらに、プロジェクト中のアーカイブごとに、アクセスできるユーザを定義します。権限は、ユーザがそのプロジェクトとアーカイブに対して実行できるアクションを定義します。

この章では、これらの機能のそれぞれについて詳細に説明します。また、この章ではセキュリティの計画、セットアップ、およびメンテナンスについても説明しています。

UNIX および Windows 上のプログラムファイルとプロジェクトデータを保護するため、設定すべき権限に関しては『Serena ChangeMan Version Manager インストールガイド』を参照してください。

Version Manager File Server セキュリティの設定に関する詳細は、次の表を参照してください。

参照内容 . .	参照先 . .
ファイルサーバセキュリティオプションの設定	142 ページの『File Server の設定』
ファイルサーバセキュリティに関する特記事項	166 ページの『セキュリティに関する特記事項』
パスマップセキュリティオプション	141 ページの『パスマップセキュリティオプションの設定』

アクセスコントロールデータベースについて

アクセスコントロールデータベースは、プロジェクトに対してアクションを実行することを認められたユーザを定義し、そのユーザが実行できるアクションを定義します。Version Manager では、アクションが Version Manager の権限に関連付けられています。

アクセスコントロールデータベースは暗号化されたファイルになっています。

ユーザ

アクセスコントロールデータベースを定義する場合、その定義には認証されたユーザごとのユーザ ID が含まれます。アクセスコントロールデータベースをセキュリティに使用している場合は、Version Manager が取得したユーザ ID がそのプロジェクトデータベースまたはプロジェクトのアクセスコントロールデータベースに定義されていなければ、そのユーザはそのプロジェクトデータベースまたはプロジェクトにアクセスできず、したがって、アーカイブにアクセスできません。Version Manager では、ユーザ ID はログインソースを使用して取得されます。ログインソースについては、90 ページの『ログインソース』を参照してください。

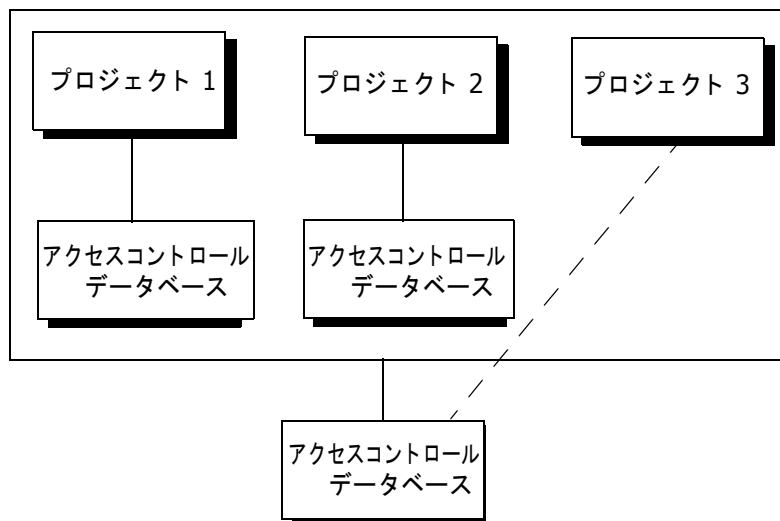
デスクトップクライアントユーザの場合：Version Manager がユーザ ID を取得すると、アクセスコントロールデータベースでそのユーザ ID が存在するかどうかチェックされます。ユーザ ID が存在しない場合は、アクセスコントロールデータベース内に自動的にユーザ ID を作成し、そのユーザにデフォルトの権限が割り当てられるように設定することができます。アクセスコントロールデータベース内に自動的にユーザが作成されるよう Version Manager を設定する手順については、[90 ページの『ログインソース』](#)を参照してください。

デスクトップクライアントの場合

デスクトップクライアントでは、アクセスコントロールデータベースはプロジェクトデータベースとプロジェクトに関連付けられています。デスクトップクライアントがアクセスコントロールデータベースを使用するのは、それがプロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル内で指定され、かつ有効に設定されている場合、またはそれが Version Manager に埋め込まれている場合です。デフォルトの動作では、プロジェクトデータベースを作成するとアクセスコントロールデータベースも作成され、同じく作成されるマスターコンフィグレーションファイル内で指定されますが、有効にはなっていません。

各プロジェクトデータベースには 1 つのアクセスコントロールデータベースを関連付けることができ、プロジェクトデータベース内の各プロジェクトにはそれぞれ 1 つのアクセスコントロールデータベースを関連付けることができます。プロジェクトデータベースに対してはアクセスコントロールデータベースが定義されているが、そのプロジェクトデータベース内のプロジェクトについては定義されていない場合は、そのプロジェクトには、プロジェクトデータベースに関連付けられたアクセスコントロールデータベースが使用されます。また、プロジェクトデータベース内のプロジェクトについて、そのうちのいくつかのみにアクセスコントロールデータベースを定義し、ほかのプロジェクトについては定義しないでおくことができます。たとえば、プロジェクトデータベース中に 3 つのプロジェクトがあり、その 3 つのうち 2 つにアクセスコントロールデータベースを定義した場合は、その 2 つのプロジェクトはそれぞれに定義されたアクセスコントロールデータベースを使用し、3 つめのプロジェクトはプロジェクトデータベースのアクセスコントロールデータベースを使用します（以下の図を参照）。

プロジェクトデータベース



プロジェクトデータベースについてアクセスコントロールデータベースを定義した場合は、そのプロジェクトデータベース内のすべてのプロジェクトに対して同一のアクセスコントロールが行われます。プロジェクトごとのコントロールをさらに厳密に行うには、各プロジェクトに対してアクセスコントロールデータベースを定義します。またアクセスリストを使用すれば、プロジェクト中のアーカイブごとにアクセスをコントロールできます。アクセスリストについては次のセクションで説明します。

プロジェクトデータベースに対するデフォルトのアクセスコントロールデータベース

Version Manager のデフォルトでは、作成されるプロジェクトデータベースごとに、デフォルトのアクセスコントロールデータベースが作成されます。このアクセスコントロールデータベースには、SuperUser 権限を割り当てられたユーザのユーザ ID と、いくつかのデフォルトの権限が含まれます (243 ページの表 6-3 : デフォルト権限セットを参照)。このアクセスコントロールデータベースは、プロジェクトデータベースのコンフィグレーションファイル中で指定されていますが、有効にはなっていません。したがって、このアクセスコントロールデータベースはこのプロジェクトデータベースのセキュリティとして使用されてはいません。

このデフォルトのアクセスコントロールデータベースは、default.db ファイルをコピーして作成されます。デフォルトではこのファイルは次の場所に配置されています。

- Windows の場合 : `drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合 : `/usr/serena/vm/common/pvcsprop/pvcs/vm`

default.db ファイルの内容は必要に応じて変更でき、Version Manager が新規アクセスコントロールデータベースを作成するときには、この変更されたファイルが使用されます。

プロジェクトデータベースに対するデフォルトのアクセスコントロールデータベースでは、そのプロジェクトデータベースへのアクセス権を持つユーザと、それらのユーザが実行することを認めるアクションを定義します。アクセスコントロールデータベース中に定義できるのはユーザ ID、パスワード、ユーザの有効期限、ユーザグループ、およびそのユーザに割り当てられた権限です。必ず定義しなければならないのはユーザ ID のみです。ユーザに権限を割り当てない場合には、後述する Unlimited 権限セットが割り当てられます。

アクセスコントロールデータベースの埋め込み

Version Manager にはアクセスコントロールデータベースを 1 つ埋め込むことができ、このアクセスコントロールデータベースがすべてのプロジェクトデータベースに関するセキュリティを制御します。アクセスコントロールデータベースを Version Manager に埋め込むことにより、すべてのユーザが同一のセキュリティ定義を使用し、ほかのアクセスコントロールデータベースは使用しないことが保証されます。

Version Manager は、埋め込まれたアクセスコントロールデータベースについては、自動的にユーザを作成することはできません (202 ページの『ユーザ』を参照)。

Version Manager に埋め込まれたアクセスコントロールデータベースは、その Version Manager を使用する、すべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。アクセスコントロールデータベースを埋め込む手順については、227 ページの『Version Manager へのアクセスコントロールデータベースの埋め込み』を参照してください。

コマンドラインインターフェイスの場合

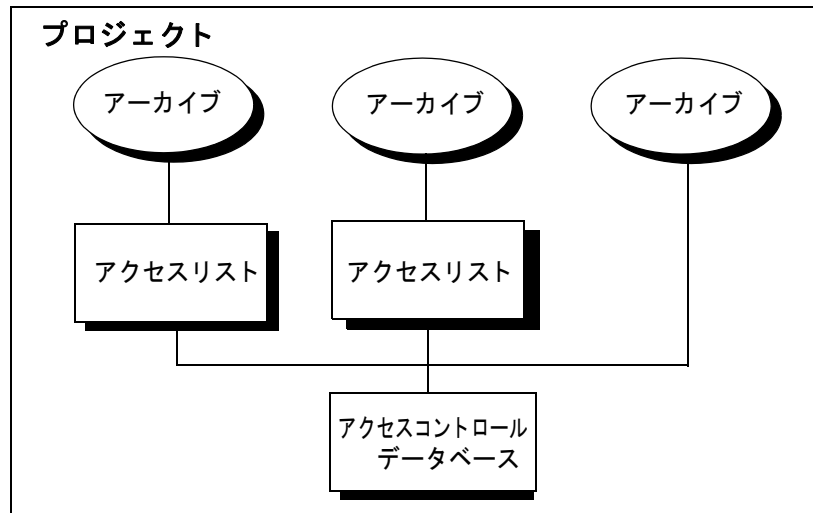
コマンドラインインターフェイスがアクセスコントロールデータベースを使用するのは、そのコマンドラインインターフェイスが使用するコンフィグレーションファイル中でそれが指定されている場合、または Version Manager に埋め込まれている場合です。アクセスコントロールデータベースを埋め込む手順については、227 ページの『Version Manager へのアクセスコントロールデータベースの埋め込み』を参照してください。

デスクトップクライアントの場合とは異なり、コマンドラインインターフェイスではアクセスコントロールデータベース中に自動的にユーザを作成することはできません。

アクセスリストについて

「アクセスリスト」は、アーカイブにアクションを実行できる権限のある個々のユーザまたはユーザグループを定義します。各アーカイブは、関連するアクセスリストを 1 つずつ持つことができます。アーカイブにアクセスリストを定義する前に、アクセスコントロールデータベースを定義する必要があります。

アクセスリストは、アクセスコントロールデータベース中に定義されたユーザのサブセットです。アクセスコントロールデータベース中に定義されていないユーザは、アクセスリスト中には定義できません。アーカイブにアクセスリストが定義されていない場合は、アクセスコントロールデータベースを使用してアーカイブへのアクセスが管理されます（次の図を参照）。



アクセスリストの使用により、セキュリティレベルが非常に高いシステムを構築できますが、アクセスリストにユーザグループでなく個々のユーザを定義すると、メンテナンスに非常に時間がかかります。したがって、アクセスリストには、個々のユーザでなくユーザグループを定義することをお勧めします。

アクセスリストグループ

アクセスリストグループの定義により、アーカイブへのアクセスをまとめて管理することができます。グループのメンバーは、プロジェクトにおいて同一の担当責任を持ち、同一の権限を持っていることが必要です。

グループ指定してアクセスリストを定義しないかぎり、グループの定義はセキュリティのために使用されません。アクセスリスト中でグループを使用するには、まずグループを定義し、そのグループに対してアクセスコントロールデータベース中で権限を割り当てる必要があります。

グループを使用することの利点は、アクセスリストの保守が簡単になることです。たとえば、開発部門に属するユーザとその権限を定義するグループを作成し、その後に新しい開発者が入社した場合は、この社員をこのグループのユーザとして追加するだけで、この社員に関する設定が完了し、この新規ユーザが、このグループがアクセスできるすべてのアーカイブにアクセスできるようになります。グループを使用しないで、アーカイブへのアクセス権をこの新規ユーザに与える場合は、このユーザをそれぞれのアーカイブのアクセスリストに追加しなければなりません。

アクセスリストグループに関しては、次の 3 つの重要なルールがあります。

- グループ定義は、そのグループがアーカイブのアクセスリスト中に定義されているときのみ使用されます。

- ユーザの権限は、グループに割り当てられた権限により、制限されますが、拡張されることはありません（207 ページの『権限のルール』を参照）。
- アクセスリストグループを空のままにしておくことはできません。アクセスリストグループの定義には、ユーザまたはアクセスリストグループが含まれている必要があります。

権限について

権限により、Version Manager アクションを実行できるかどうかが決まります。権限は、アクセスコントロールデータベース中で定義され、ユーザにアクションの実行を許可します。たとえば、「チップリビジョンのチェックアウト」という権限は、アーカイブからチップリビジョンをチェックアウトすることを許可します。

Version Manager には次の 3 種類の権限があります。

- ベース権限
- コンポジット権限
- 権限セット

Version Manager に組み込まれている権限を使用することも、必要に応じてカスタム権限を定義することもできます。

権限は、アクセスコントロールデータベース中で、個別のユーザまたはユーザグループ（アクセスリストグループ）に対して割り当てられます。

デスクトップクライアントでは、機能に応じて次のように権限がグループ分けされます。

- アクション
- アーカイブ
- Project（プロジェクト）



注 権限が関連づけられていないおもなアクションの 1 つに、プロジェクトデータベースの作成があります。ユーザによるプロジェクトデータベースの作成を制限するには、ユーザがこのアクションを実行するときに入力しなければならないパスワードを定義する必要があります。プロジェクトデータベースの作成を許可しないユーザには、パスワードを知らせないようにしてください。詳しい説明は、228 ページの『プロジェクトデータベース作成の制限』にあります。

ベース権限

ベース権限とは、単独の Version Manager アクションです。たとえば、AddVersion 権限を持つユーザは、リビジョンにバージョンラベルを割り当てることができます。ベース権限のリストについては、239 ページの表 6-1：ベース権限を参照してください。

コンポジット権限

コンポジット権限は、2 つまたはそれ以上のベース権限を組み合わせたものです。たとえば、「チェックアウト」コンポジット権限は、ベース権限の「チップ以外のリビジョンのチェックアウト」と「チップリビジョンのチェックアウト」を組み合わせたものです。この権限があると、チップとチップ以外の両方のリビジョンをアーカイブからチェックアウトできます。コンポジット権限のリストについては、242 ページの表 6-2：コンポジット権限を参照してください。

権限セット

権限セットはベース権限、コンポジット権限、およびほかの権限セットを組み合わせたものです。Version Manager に設定されたデフォルトの権限セットは、243 ページの表 6-3 : デフォルト権限セットに定義されています。

Version Manager では作業環境に応じた権限セットを作成できるよう、権限セットの定義をカスタマイズすることができます。たとえば、コンポジット権限の「チェックアウト」と「チェックイン」、およびベース権限の「アーカイブの作成」から、BasicUser という名称のカスタム権限セットを作成できます。BasicUser を割り当てられたユーザは、リビジョンのチェックインとチェックアウト、およびアーカイブの作成を行うことができます。

カスタム権限セットを新しく定義するときは、既存の権限セット（デフォルトおよびカスタム）をその定義の一部として使用できます。

権限のルール

Version Manager の権限には、次のルールが適用されます。

- デフォルトでは、ユーザに Unlimited 権限が与えられています。また、アクセスコントロールデータベースにおいてユーザに Unlimited 権限が与えられている場合は、アーカイブへのアクセス権はアクセスリストによってのみ制限されます。
- アクセスコントロールデータベースにおいてユーザに SuperUser 権限が与えられている場合は、アーカイブのアクセスリストにそのユーザが存在するかどうかにかかわらず、そのアーカイブへのアクセスは制限されません。
- アクセスリストグループに割り当てられた権限は、そのグループに属するユーザの権限を制限するようにのみ働き、ユーザの権限を拡張することはありません。たとえば、JohnD というユーザは Unlimited 権限セットが割り当てられており、「開発者」権限セットが割り当てられた「開発者」グループに属するとします。「開発者」権限セットに与えられた権限はロック、ロック解除、チェックアウト、チェックインです。JohnD がアクセスしようとするアーカイブのアクセスリストには「開発者」グループはありますが、個別ユーザとしての JohnD は存在しないとします。この場合、JohnD に与えられる権限はロック、ロック解除、チェックアウト、チェックインで、与えられていたその他の権限は使用できません。これは、このアーカイブのアクセスリストに「開発者」グループは存在しているが、JohnD は存在しないためです。

アーカイブのアクセスリストに「開発者」グループと JohnD の両方が存在する場合は、JohnD には Unlimited 権限が与えられます。

逆に、JohnD に「開発者」権限セットが割り当てられており、Unlimited 権限セットが割り当てられた「開発者」グループに属するとします。JohnD がアクセスしようとするアーカイブのアクセスリストに「開発者」グループのみが存在する場合は、JohnD に与えられる権限は「開発者」権限（「ロック」、「ロック解除」、「チェックアウト」、および「チェックイン」）のみとなります。「開発者」グループに割り当てられた Unlimited 権限セットは、個別ユーザとして JohnD に割り当てられているのではないため、JohnD には与えられません。

- アクセスリストグループの定義には、個別のユーザだけでなくほかのグループも含めることができます。個別のユーザが複数のグループに属し、そのユーザが属する各グループがアクセスリストグループの定義に含まれている場合は、そのユーザには各グループに割り当てられた権限の和集合が与えられます。
- SuperUser はグループに割り当てることができず、個別のユーザにのみ割り当てられます。

セキュリティの計画

以下は、セキュリティを計画するときを目安となる項目です。

- 必要なセキュリティレベル。
 - プロジェクトデータベース中の各プロジェクトに対し、同一のセキュリティ定義を使用できる可能性。同一のセキュリティ定義を使用できる場合はプロジェクトデータベースに対してアクセスコントロールデータベースを定義することができます。プロジェクトデータベース内の各プロジェクトに対して定義する必要はなく、アクセスリストも必要ありません。また、各ユーザには同一の権限を設定できます（後述の例 1 を参照してください）。
 - プロジェクトデータベース中の各プロジェクトに対して、同一のセキュリティ定義を使用できる可能性。この場合は、プロジェクトデータベースに対してアクセスコントロールデータベースを定義することができます。各プロジェクトに対して定義する必要はなく、アクセスリストも必要ありません。また、ユーザがそれぞれのアーカイブに対して異なる役割を持つ場合、それぞれのユーザの役割に対して権限セットを定義し、各ユーザに該当する権限セットを割り当てることができます（後述の例 2 を参照してください）。
 - プロジェクトデータベース中の各プロジェクトのアーカイブについて、異なるセキュリティを定義する必要性の有無。異なる定義が必要な場合は、各プロジェクトについて、アクセスリストを定義せずに、アクセスコントロールデータベースを定義することができます。
 - プロジェクトデータベース中のアーカイブ中での、異なるセキュリティを定義する必要性の有無。このようなアーカイブが含まれる場合は、それらに対してアクセスリストを定義する必要があります。これにより、最高レベルのセキュリティが提供されます（後述の例 3 を参照してください）。
- アーカイブへのアクセスをまとめて管理するため、アクセスリストグループを定義する必要性の有無。まとめて管理するには、アクセスリストを使用する必要があります。アクセスリストを定義する場合は、アクセスリストグループを使用してください（205 ページの『アクセスリストについて』を参照してください）。

例

このセクションでは、セキュリティの設定時にモデルとなる事例を示します。

例 1

プロジェクトデータベース外のユーザからアーカイブにアクセスできないよう、プロジェクトデータベースに関してセキュリティを定義するとします。すべてのユーザには同一の権限セットを割り当てます。

アクセスコントロールデータベース：プロジェクトデータベースにアクセスする必要のあるすべてのユーザを定義し、それらのユーザに与えるカスタム権限セットを定義するか、デフォルトの権限セットを選択し、それを各ユーザに割り当てます。

アクセスリスト：なし。したがって、アクセスリストグループは必要ありません。

例 2

プロジェクトデータベース外のユーザからアーカイブにアクセスできないよう、プロジェクトデータベースに関してセキュリティを定義するとします。このケースでは、役割に関連した権限を割り当てます。たとえば、開発者には開発関連の権限セットを割り当て、QA エンジニアには別の権限セット、プロジェクトリーダーにはさらに別の権限セット、というように割り当てを行います。

アクセスコントロールデータベース：プロジェクトデータベースにアクセスする必要があるすべてのユーザを定義し、それらのユーザに与えるカスタム権限セットを定義するか、デフォルトの権限セットを選択し、それを各ユーザに割り当てます。Version Manager のデフォルト権限セットは、企業内の役割によるグループに合わせて定義されています（243 ページの表 6-3：デフォルト権限セットを参照してください）。

アクセスリスト：なし。したがって、アクセスリストグループは必要ありません。

例 3

この例は例 2 に基づいて作成されています。機密ファイルを含むアーカイブファイルについて、個々にセキュリティを設定する必要があるとします。たとえば、ほとんどの企業では、給与に関する情報には限定されたユーザにのみアクセスを許可しています。この例では、役割に関連した権限を割り当てただけでなく、アーカイブリストを機密アーカイブに割り当てます。

アクセスコントロールデータベース：プロジェクトデータベースにアクセスする必要があるすべてのユーザを定義し、それらのユーザに与えるカスタム権限セットを定義するか、デフォルトの権限セットを選択し、それを各ユーザに割り当てます。Version Manager のデフォルト権限セットは、企業内の役割によるグループに合わせて定義されています（243 ページの表 6-3：デフォルト権限セットを参照してください）。

また、制限されたアーカイブへのアクセスを許可される個々のユーザを含むアクセスリストグループを定義し、そのグループに該当する権限または権限セットを割り当てます。このアクセスリストグループの定義には、アクセスコントロールデータベース中で定義された個々のユーザのみを含められます。

アクセスリスト：機密アーカイブに割り当てられるもの。アクセスコントロールデータベース中で定義したアクセスリストグループが、このアクセスリストの定義になります。

セキュリティの設定

このセクションでは、Version Manager デスクトップクライアントおよびコマンドラインインターフェイスを使用してセキュリティを設定するための手順を順に説明します。

デスクトップクライアントを使用する場合

セキュリティを設定するための基本的手順を次に示します。詳細な手順については、以降の各セクションで説明しています。

- 1 プロジェクトに対してセキュリティを設定するには、プロジェクトコンフィグレーションファイルとそのプロジェクトに関連付けます。すでに関連付けられている場合には、この作業は必要ありません。次にアクセスコントロールデータベースの場所を指定するか、このプロジェクト用に新しく作成します。
- 2 プロジェクトデータベースに対するセキュリティの設定において、そのプロジェクトデータベースとともに作成されたもの以外のアクセスコントロールデータベースを使用する場合は、そのアクセスコントロールデータベースの場所を指定するか、新しく作成します。
- 3 アクセスコントロールデータベースに対するカスタム権限セットを定義します。
- 4 ユーザおよびそのユーザに割り当てる権限を定義します。
- 5 セキュリティ定義の一部としてアクセスリストを使用する場合は、グループに権限とユーザを割り当てて、適切なアクセスリストグループを定義します。

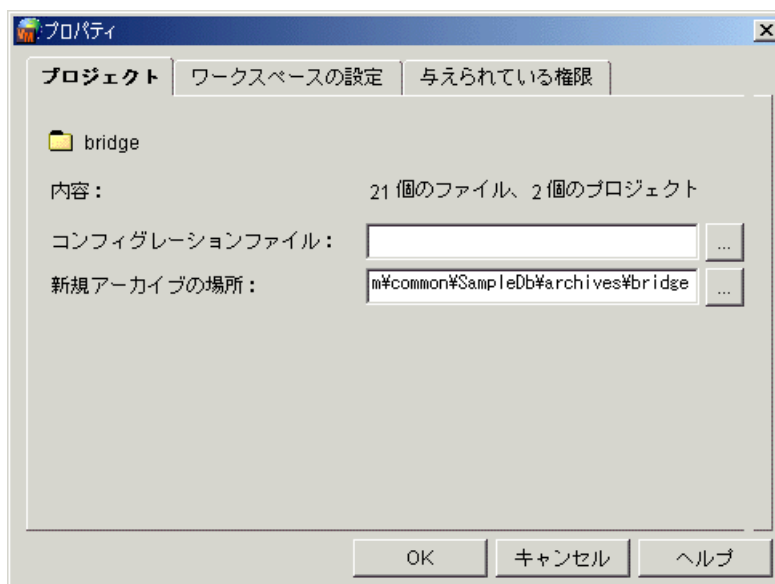
- 6 アクセスリストを使用する場合は、該当するアーカイブに対するアクセスリストを定義します。
- 7 セキュリティを有効にします。

プロジェクトへのコンフィグレーションファイルの関連付け

プロジェクトは、デフォルトではプロジェクトコンフィグレーションファイルには関連付けられていません。

プロジェクトへのプロジェクトコンフィグレーションファイルを関連付けるには

- 1 コンフィグレーションファイルに関連付けるプロジェクトを選択します。
- 2 [ファイル] - [プロパティ] を選択します。[プロパティ] ダイアログボックスが表示され、[プロジェクト] タブがアクティブになっています。



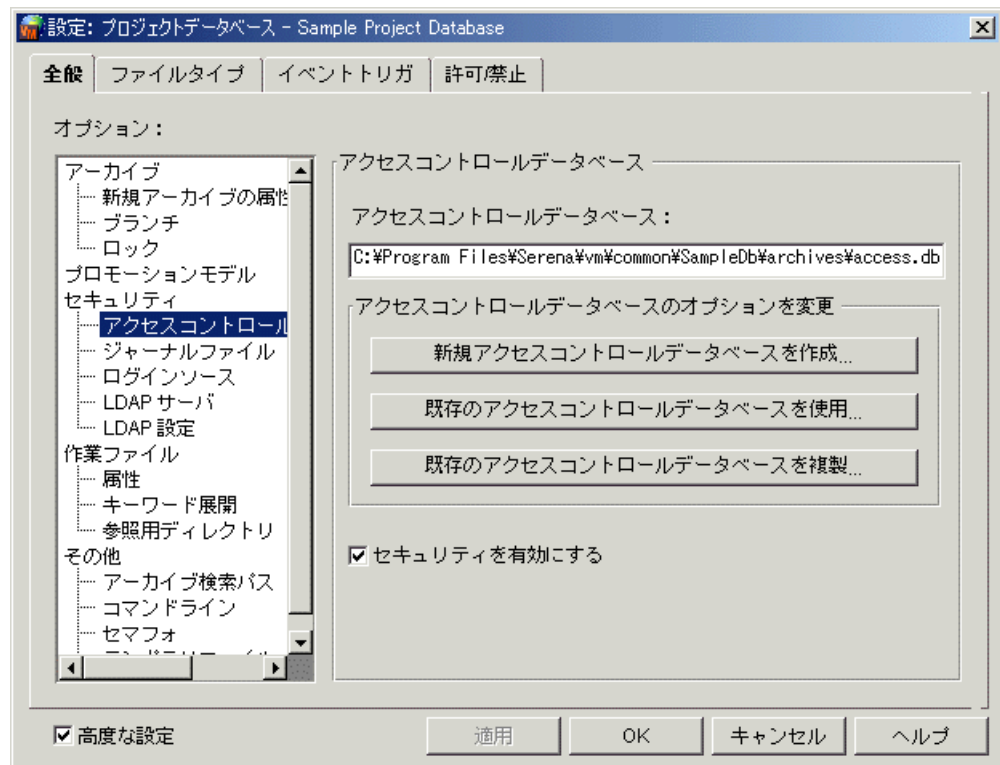
- 3 〈コンフィグレーションファイル〉テキストボックスに、プロジェクトに関連付けるコンフィグレーションファイルの場所と名前を入力します。コンフィグレーションファイルが存在しない場合は、Version Manager がここで指定した場所と名前を使用して自動的に作成します。デフォルトの名前は project.cfg です。このコンフィグレーションファイルでは、オプションは設定されていません。
- 4 【OK】をクリックします。

プロジェクトデータベースまたはプロジェクトに対するアクセスコントロールデータベースの指定

アクセスコントロールデータベースの場所を指定するには

- 1 アクセスコントロールデータベースを定義する、対象となるプロジェクトデータベースまたはプロジェクトを選択します。このプロジェクトデータベースまたはプロジェクトには、コンフィグレーションファイルが関連付けられていることが必要です。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。

- 4 〈オプション〉リストから [セキュリティ] - [アクセスコントロールデータベース] を選択します。このアクセスコントロールデータベースに関する情報が右側の〈アクセスコントロールデータベース〉表示部に表示されます。



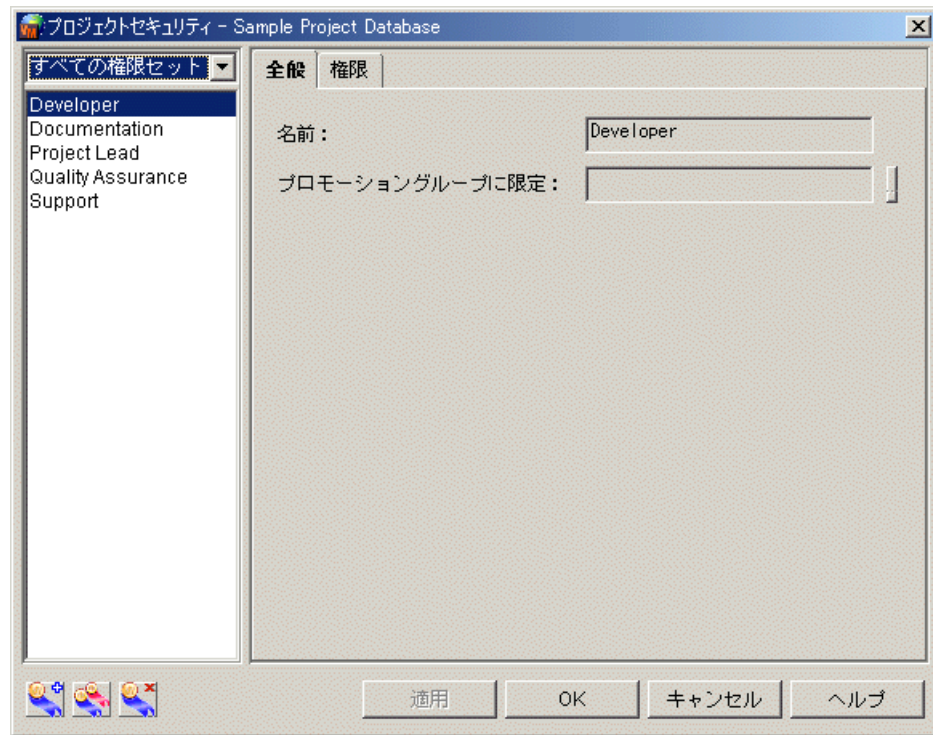
- 5 次のいずれかの作業を行います。
- 【新規アクセスコントロールデータベースを作成】をクリックし、プロジェクトデータベースまたはプロジェクトのアーカイブディレクトリに、新しいアクセスコントロールデータベースを作成します。デフォルトのアクセスコントロールデータベース (default.db) がこの場所にコピーされます。
 - 既存のアクセスコントロールデータベースを使用するには、【既存のアクセスコントロールデータベースを使用】をクリックします。アクセスコントロールデータベースを選択するための [アクセスコントロールデータベースの選択] ダイアログボックスが表示されます。このオプションは、同一のセキュリティ情報を必要とする複数のプロジェクトデータベースが存在する場合に便利です。同一のアクセスコントロールデータベースをそれぞれのプロジェクトデータベースに使用していれば、セキュリティの必要が変化したとき、たとえばアクセスコントロールデータベースに新規ユーザを追加するときなどに、1 つのアクセスコントロールデータベースを更新するだけで済みます。
 - 既存のアクセスコントロールデータベースを複製するには、【既存のアクセスコントロールデータベースを複製】をクリックします。アクセスコントロールデータベースを選択するための [アクセスコントロールデータベースの選択] ダイアログボックスが表示されます。Version Manager は選択したアクセスコントロールデータベースのコピーを作成し、それをプロジェクトデータベースのアーカイブディレクトリに配置します。
- 6 アクセスコントロールデータベースをその内容を適用することなく修正できるよう、〈アクセスコントロールデータベースのセキュリティを有効にする〉チェックボックスの選択を解除します。
- 7 【OK】をクリックします。

カスタム権限セットの定義

アクセスコントロールデータベースに対するカスタム権限セットを定義します。

カスタム権限セットを定義するには

- 1 アクセスコントロールデータベースを定義する対象となるプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [権限セット] を選択します。[プロジェクトセキュリティ] ダイアログボックスが開き、[全般] タブがアクティブになっています。





- 3 [新規作成] アイコンをクリックします。[権限セットの作成] ダイアログボックスが表示されます。

- 4 カスタム権限セットの名前を入力します。名前の最初および最後の文字に、タブやスペースは使用できません。使用可能な文字は、半角のかっこ (()), 引用符 ('), 二重引用符 ("), コロン (:), 円記号 (¥), アスタリスク (*) を除くすべての文字です。
- 5 この新規カスタム権限セットを定義するベース権限、コンポジット権限、および既存の権限セットを選択します。すべての権限のリストと定義については、[238 ページの『権限の定義』](#)を参照してください。

隣にあるボックスに「+」の付いた権限は有効になっています。「-」は、禁止権限であるプロジェクト権限のみに付けられ、その権限が禁止されていることを示します。プロジェクト権限がユーザに対して禁止されている場合は、この権限をそのユーザに割り当てられた権限セットで許可することはできません。たとえば、「プロジェクトの作成」権限がユーザに対するベース権限として禁止されているが、そのユーザに対する権限セットでは許可されている場合は、このユーザにはプロジェクトの作成権限は与えられません。

権限の選択状況を変更するには、権限名の隣のチェックボックスを選択します。親権限をクリックすることにより、関連する権限のセット全体で有効または無効にすることができます。たとえば、「アクション権限」の隣のチェックボックスを選択して「+」記号を表示することにより、すべてのアクション権限を割り当てることができます。

ダイアログボックス下部の〈選択された権限〉には、選択された権限がすべて表示されます。プロジェクト権限のいずれかが選択されている場合は、このボックスには表示されません。プロジェクト権限は禁止権限のため、無効に設定されている場合のみこのボックスに表示されます。

- 6 〈権限をプロモーショングループに限定〉テキストボックスに、カスタム権限セットを制限するためのプロモーショングループを入力します。[256 ページの『ユーザによるプロモートの制限』](#)を参照してください。

- 7 【作成】をクリックしてこの権限セットを作成します。
- 8 さらに権限セットを定義するには手順 4～7 を繰り返します。定義しない場合は、【閉じる】をクリックします。【閉じる】をクリックすると、[権限セットの作成] ダイアログボックスが閉じ、新規に作成された権限セットが [プロジェクトセキュリティ] ダイアログボックスの〈すべての権限セット〉リストに追加されて選択されます。
- 9 次の手順 **ユーザの定義**に進みます。

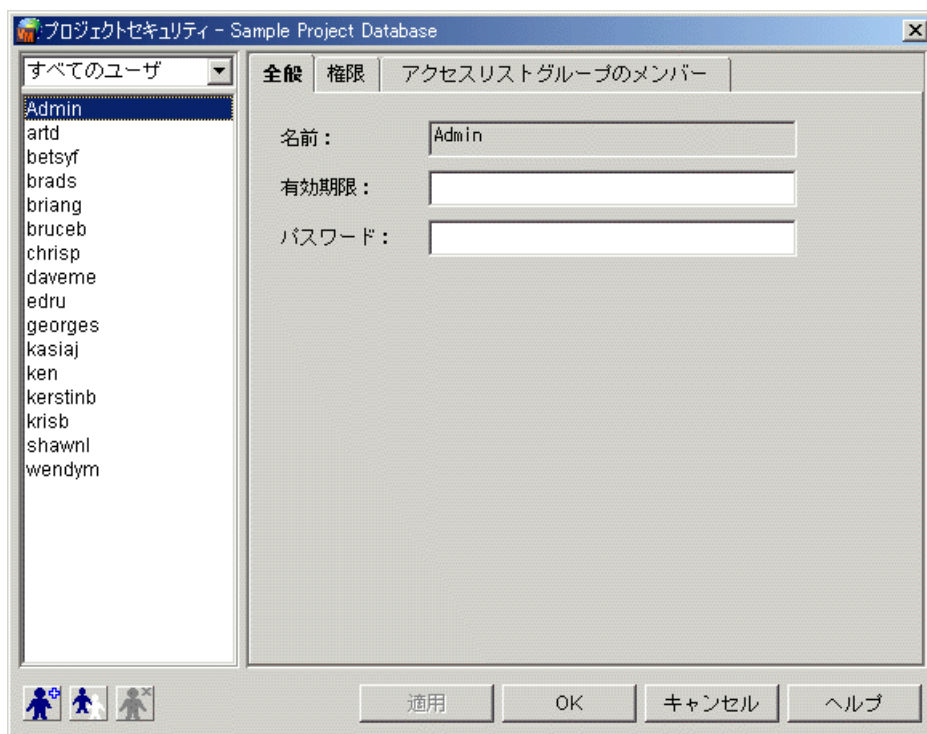
ユーザの定義

デスクトップクライアントユーザの場合 : Version Manager がユーザ ID を取得すると、アクセスコントロールデータベースでそのユーザ ID が存在するかどうかチェックされます。ユーザ ID が存在しない場合は、アクセスコントロールデータベース内に自動的にユーザ ID を作成し、そのユーザに権限が割り当てられるように設定することができます。アクセスコントロールデータベース内に自動的にユーザが作成されるよう Version Manager を設定する手順については、[90 ページの『ロギンソース』](#)を参照してください。

アクセスコントロールデータベース中にユーザを定義します。これには、ユーザへの権限の付与も含まれます。

アクセスコントロールデータベース中にユーザを定義するには

- 1 [プロジェクトセキュリティ] ダイアログボックスがすでに開いている場合は、左上のドロップダウンリストから [すべてのユーザ] を選択します。開いていない場合は [管理] - [セキュリティ] - [ユーザ] を選択します。次のような [プロジェクトセキュリティ] ダイアログボックスが表示されます。



- 2 [新規作成] アイコンをクリックします。[ユーザの作成] ダイアログボックスが表示されます。

- 3 <名前> フィールドにユーザ ID を入力します。ここに入力するユーザ ID は、このユーザが Version Manager を実行するときに使用されるものであることが必要です。Version Manager のユーザ ID の取得方法については、90 ページの『ログインソース』を参照してください。
- 4 必要に応じて、<有効期限> テキストボックスにユーザの有効期限を入力できます。たとえば、2003 年 1 月 2 日にユーザを定義し、有効期日に 2003 年 12 月 31 日を入力すると、このユーザ ID は 2003 年の 1 月 2 日から 12 月 31 日までの期間に有効となります。入力する日付と時刻のフォーマットは、オペレーティングシステムで設定されているものを使用します。Windows では、日付と時刻が [コントロールパネル] の [地域] で設定されています。UNIX では、PVCS_DATE_FORMAT および PVCS_TIME_FORMAT 環境変数を設定することにより定義されます。コントロールパネルや環境変数による設定が行われていない場合は、mm/dd/yy hh:mm のフォーマットが使用されます。



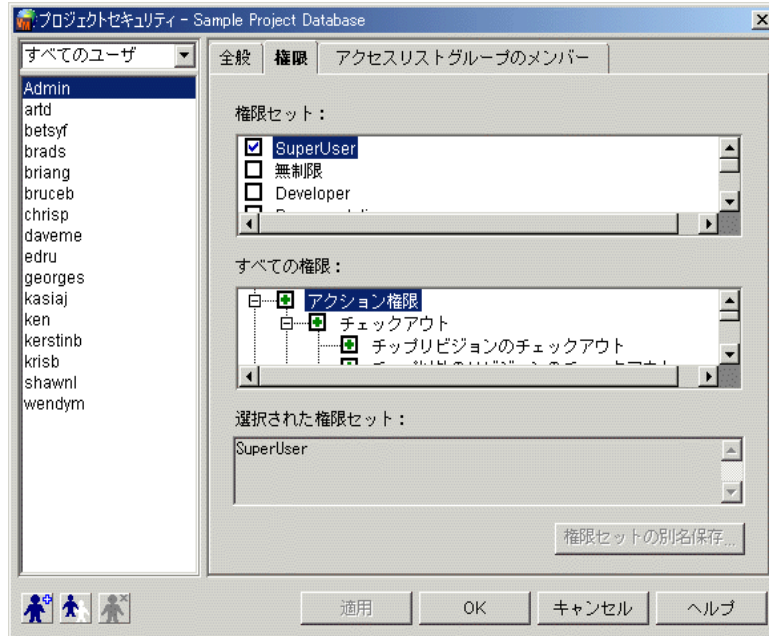
注 UNIX 上では、PVCS_DATE_FORMAT を MM/dd/yyyy (日を先にする場合は dd/MM/yyyy) に設定することをお勧めします。ここで、d と y はそれぞれ、日と年を表す数字です。M は月を表す英語の略号です。年は 4 桁で、月は大文字で指定してください。このフォーマットを使用すると、最適な結果が得られます。

- 5 ユーザが、アクセスコントロールデータベースが設定されたプロジェクトデータベースまたはプロジェクトにアクセスする前にパスワードの入力を要求するには、パスワードを <パスワード> テキストボックスに、半角英数字 (記号を含む) 30 字以内で入力します。[ログイン] ダイアログボックスがログインソースとして指定され、Version Manager がこのログインソースを使用してユーザ ID を取得する場合は、ユーザは自分に設定されたユーザ ID とパスワードを入力する必要があります。ユーザ ID が別のログインソースを使用して入手され、ユーザにパスワードが設定されている場合は、ユーザにはパスワードの入力のみが要求されます。

ユーザは [ファイル] - [パスワードの変更] を選択して、いつでも自分のパスワードを変更できます。

- 6 【作成】をクリックし、次に【閉じる】をクリックしてこのユーザをアクセスコントロールデータベース中に作成します。【閉じる】をクリックすると [ユーザの作成] ダイアログボックスが閉じ、作成されたユーザは [プロジェクトセキュリティ] ダイアログボックスの <すべてのユーザ> リストに追加されて、選択された状態になります。

7 [権限] タブをクリックします。



8 定義するユーザが〈すべてのユーザ〉リスト中で選択されていることを確認します。

9 [権限] タブ上でユーザに割り当てる権限または権限セット、またはそれらの組合せを選択します。権限のリストと定義については、238 ページの『権限の定義』を参照してください。

隣にあるボックスに「+」の付いた権限は有効になっています。「-」は、禁止権限であるプロジェクト権限のみに付けられ、その権限が禁止されていることを示します。プロジェクト権限がユーザに対して禁止されている場合は、この権限をそのユーザに割り当てられた権限セットで許可することはできません。たとえば、「プロジェクトの作成」権限がユーザに対するベース権限として禁止されているが、そのユーザに対する権限セットでは許可されている場合は、このユーザにはプロジェクトの作成権限は与えられません。

権限の選択状況を変更するには、権限名の隣のチェックボックスをクリックします。親権限をクリックすることにより、関連する権限のセット全体で有効または無効にすることができます。たとえば、「アクション権限」の隣のチェックボックスをクリックして「+」記号を表示することにより、すべてのアクション権限を割り当てることができます。

ダイアログボックス下部の〈選択された権限〉には、選択された権限がすべて表示されます。プロジェクト権限のいずれかが選択されている場合は、このボックスには表示されません。プロジェクト権限は禁止権限のため、無効に設定されている場合にのみこのボックスに表示されます。

10 ユーザに割り当てた権限を保存し、さらに別のユーザの権限を定義するには、【適用】をクリックし、手順 11 に進みます。これでユーザ権限の定義を終了する場合は、【OK】をクリックします。

11 さらにユーザの権限を定義するには、次のいずれかの作業を行います。



- [ユーザの作成] アイコンをクリックし、手順 3 ~ 10 を繰り返して新規ユーザを作成します。



- ユーザを選択し、[ユーザの複製] アイコンをクリックして、既存のユーザを複製します。[ユーザの複製] ダイアログボックスが表示されます。この複製機能は同一の権限セットを割り当てるユーザを複数定義する場合に便利です。

[ユーザの複製] ダイアログボックス内で次の作業を行います。

- 〈名前〉フィールドにユーザのユーザ ID を入力します。
- 必要に応じて、〈有効期限〉テキストボックスにユーザの有効期限を入力できます。たとえば、2003 年 1 月 2 日にユーザを定義し、有効期日に 2003 年 12 月 31 日を入力すると、このユーザ ID は 2003 年の 1 月 2 日から 12 月 31 日までの期間に有効となります。入力する日付と時刻のフォーマットは、オペレーティングシステムで設定されているものを使用します。Windows では、日付と時刻が [コントロールパネル] の [地域] で設定されています。UNIX では、PVCS_DATE_FORMAT および PVCS_TIME_FORMAT 環境変数を設定することにより定義されます。コントロールパネルや環境変数による設定が行われていない場合は、mm/dd/yy hh:mm のフォーマットが使用されます。



注 UNIX 上では、PVCS_DATE_FORMAT を MM/dd/yyyy (日を先にする場合は dd/MM/yyyy) に設定することをお勧めします。ここで、d と y はそれぞれ、日と年を表す数字です。M は月を表す英語の略号です。年は 4 桁で、月は大文字で指定してください。このフォーマットを使用すると、最適な結果が得られます。

- ユーザが、アクセスコントロールデータベースが設定されたプロジェクトデータベースまたはプロジェクトにアクセスする前にパスワードの入力を要求するには、パスワードを〈パスワード〉テキストボックスに、半角英数字 (記号を含む) 30 字以内で入力します。[ログイン] ダイアログボックスがログインソースとして指定され、Version Manager がこのログインソースを使用してユーザ ID を取得する場合は、ユーザは自分に設定されたユーザ ID とパスワードを入力する必要があります。ユーザ ID が別のログインソースを使用して入手され、ユーザにパスワードが設定されている場合は、ユーザにはパスワードの入力のみが要求されます。

ユーザは [ファイル] - [パスワードの変更] を選択して、いつでも自分のパスワードを変更できます。

- 〈権限のコピー〉チェックボックスを選択し、コピー元のユーザに対して定義されていた権限をすべて新しいユーザに割り当てます。
- この手順ではグループを定義していないので、〈メンバーのコピー〉チェックボックスから選択を解除します。
- 【複製】をクリックします。
- さらにユーザを複製するには手順 11a ~ f を繰り返します。複製しない場合は【閉じる】をクリックします。

- 12 セキュリティ定義の一部としてアクセスリストを使用する場合は、次の手順、『アクセスリストグループの定義』に進みます。使用しない場合は、221 ページの『セキュリティの有効化』に進みます。

アクセスリストグループの定義

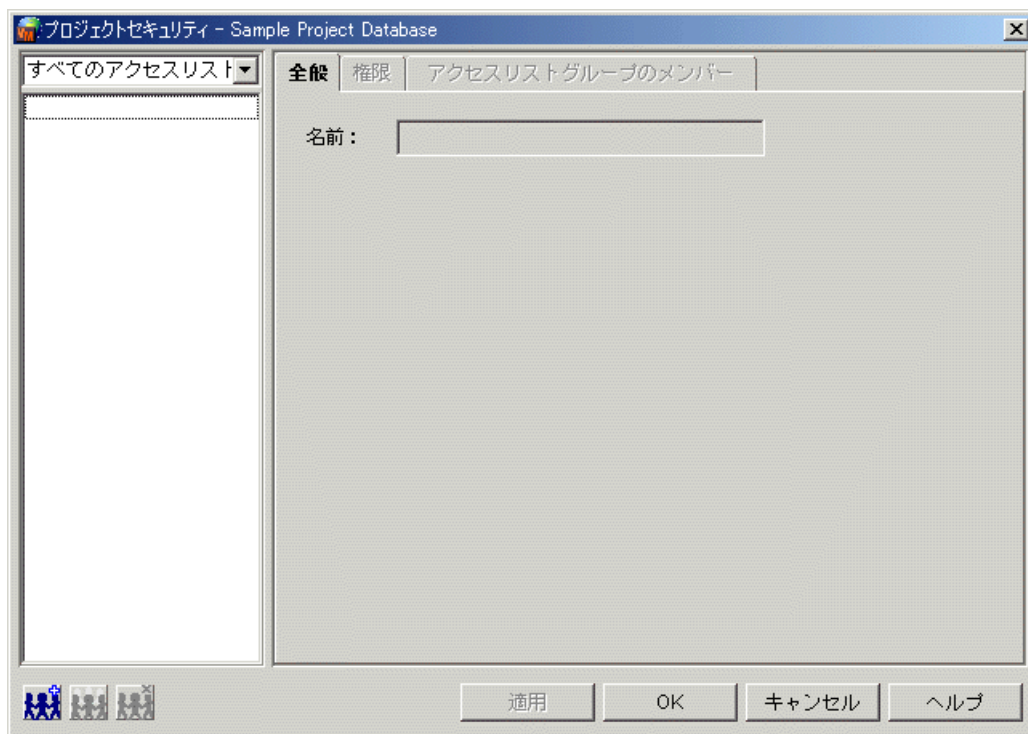
セキュリティ定義の一部としてアクセスリストを使用する場合は、グループに権限とユーザを割り当て、適切なアクセスリストグループを定義します。



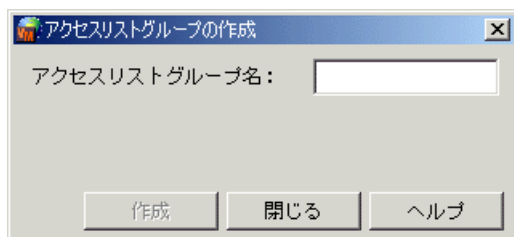
注 アクセスリストグループを作成する場合は必ず、アクセスリストグループを定義するユーザまたはほかのアクセスリストグループを指定する必要があります。アクセスリストグループは、空（から）のままにしておくことはできません。

アクセスリストグループを定義するには

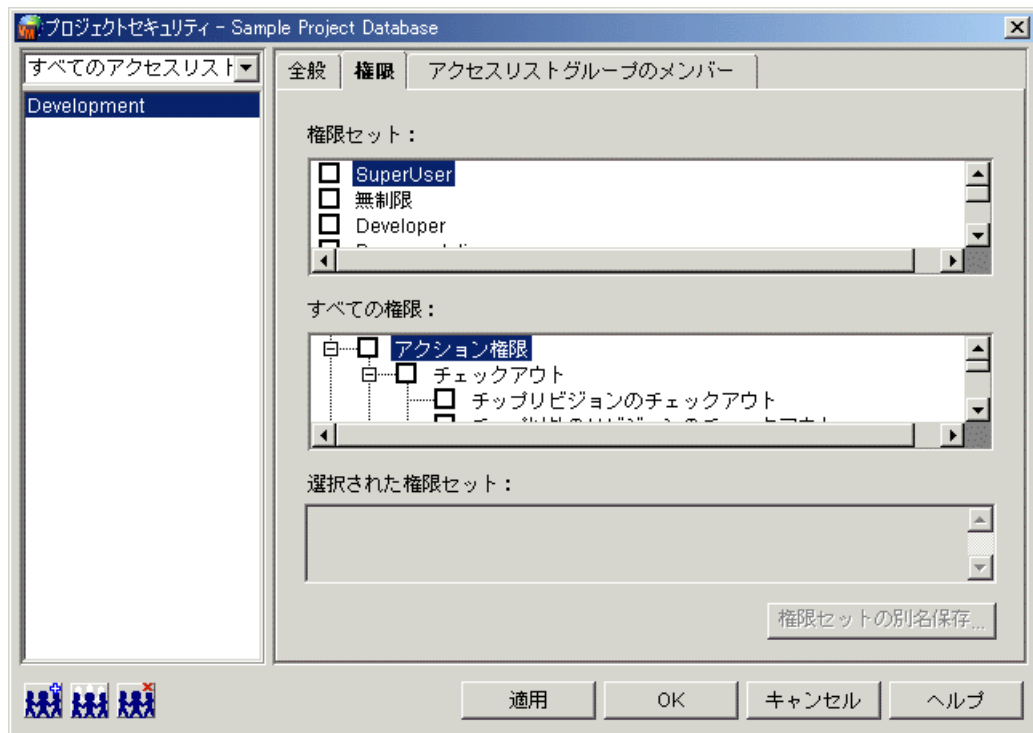
- 1 [プロジェクトセキュリティ] ダイアログボックスがすでに開いている場合は、左上のドロップダウンリストから [すべてのユーザ] を選択します。開いていない場合は、[管理] - [セキュリティ] - [アクセスリストグループ] を選択します。次のような [プロジェクトセキュリティ] ダイアログボックスが表示されます。



- 2 [新規作成] アイコンをクリックします。[アクセスリストグループの作成] ダイアログボックスが表示されます。



- 3 <名前> テキストボックスにグループの名前を入力します。名前の最初および最後の文字に、タブやスペースは使用できません。使用可能な文字は、半角のかっこ (())、引用符 (')、二重引用符 (")、コロン (:)、円記号 (¥)、アスタリスク (*) を除くすべての文字です。
- 4 【作成】をクリックします。ダイアログボックスのテキストボックスがクリアされ、新しいグループの名前を入力できます。グループの作成を終了したら【閉じる】をクリックします。[アクセスリストグループの作成] ダイアログボックスが閉じ、新しく作成されたグループは [プロジェクトセキュリティ] ダイアログボックスの <すべてのアクセスリスト> リストに追加されて、選択された状態になります。
- 5 [権限] タブをクリックします。



- 6 定義するアクセスリストグループが、<すべてのアクセスリスト> リスト中で選択されていることを確認します。
- 7 [権限] タブで、このアクセスリストグループに割り当てるベース権限、コンポジット権限、および既存の権限セットを選択します。すべての権限のリストと定義については、[238 ページの『権限の定義』](#)を参照してください。

隣に「+」のついた権限は有効になっています。「-」は、その権限が禁止されていることを示します。禁止権限となりうるのはプロジェクト権限だけであり、「-」が付いていない権限は有効になっています。権限の選択状況を変更するには、権限名の隣のチェックボックスを選択します。親権限をクリックすることにより、関連する権限のセット全体で有効または無効にすることができます。たとえば、「アクション権限」の隣のチェックボックスを選択して「+」記号を表示することにより、すべてのアクション権限を割り当てることができます。

ダイアログボックス下部の<選択された権限>には、選択された権限がすべて表示されます。プロジェクト権限のいずれかが選択されている場合は、このボックスには表示されません。プロジェクト権限は禁止権限のため、無効に設定されている場合のみこのボックスに表示されません。

- 8 [アクセスリストグループのメンバー] タブをクリックします。このタブには、アクセスコントロールデータベース中に定義されているユーザとグループがすべて表示されます。



- 9 定義するアクセスリストグループが、〈すべてのアクセスリスト〉リスト中で選択されていることを確認します。
- 10 グループに追加するユーザまたはアクセスリストグループの隣にあるチェックボックスを選択します。この手順は、必ず実行する必要があります。アクセスリストグループを空のままにしておくことはできません。



注 デスクトップクライアントでは、空のアクセスリストグループを作成することができますが、必ずユーザまたはほかのアクセスリストグループを定義するようにしてください。

- 11 【適用】をクリックします。
- 12 さらにアクセスリストグループを作成するには、手順 2 ~ 11 を繰り返します。アクセスリストグループの定義を終了したら【OK】をクリックします。

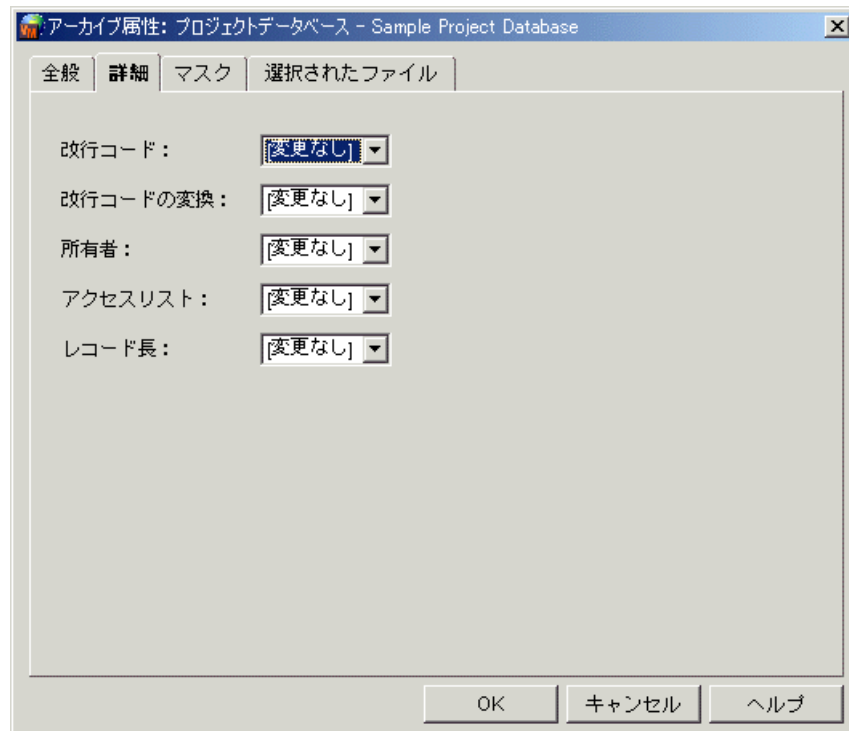
アクセスリストの定義

アクセスリストを使用する場合は、該当するアーカイブに対してアクセスリストを定義します。アクセスリストはアーカイブの属性の 1 つです。アクセスリストは新規と既存のいずれのアーカイブに対しても定義できます。新規に作成したアーカイブにアクセスリストを定義する手順については、[84 ページの『アーカイブ作成時のオプション』](#)を参照してください。

既存のアーカイブに対してアクセスリストを定義するには

- 1 アクセスリストを定義するアーカイブに関連付けられた、プロジェクトまたは個別のバージョン管理ファイルを選択します。プロジェクトを選択した場合は、プロジェクト中のすべてのアーカイブに対して同一のアクセスリストが定義されます。

- 2 [管理] - [アーカイブ属性] を選択します。[アーカイブ属性] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 [詳細] タブをクリックします。



- 4 <アクセスリスト> リストから「変更」を選択し、右側のテキストボックスに表示されるアーカイブに対してアクセスを許可する、アクセスリストグループおよび個別のユーザを入力します。リスト中の各項目はコンマ (,) で区切ります。あるいは【...】をクリックして、グループまたは個別のユーザを選択します。このテキストボックスの値は合計で 254 文字までに制限されています。

ここで指定するアクセスリストグループとユーザは、アクセスコントロールデータベース中で定義されていることが必要です。
- 5 【OK】 をクリックします。

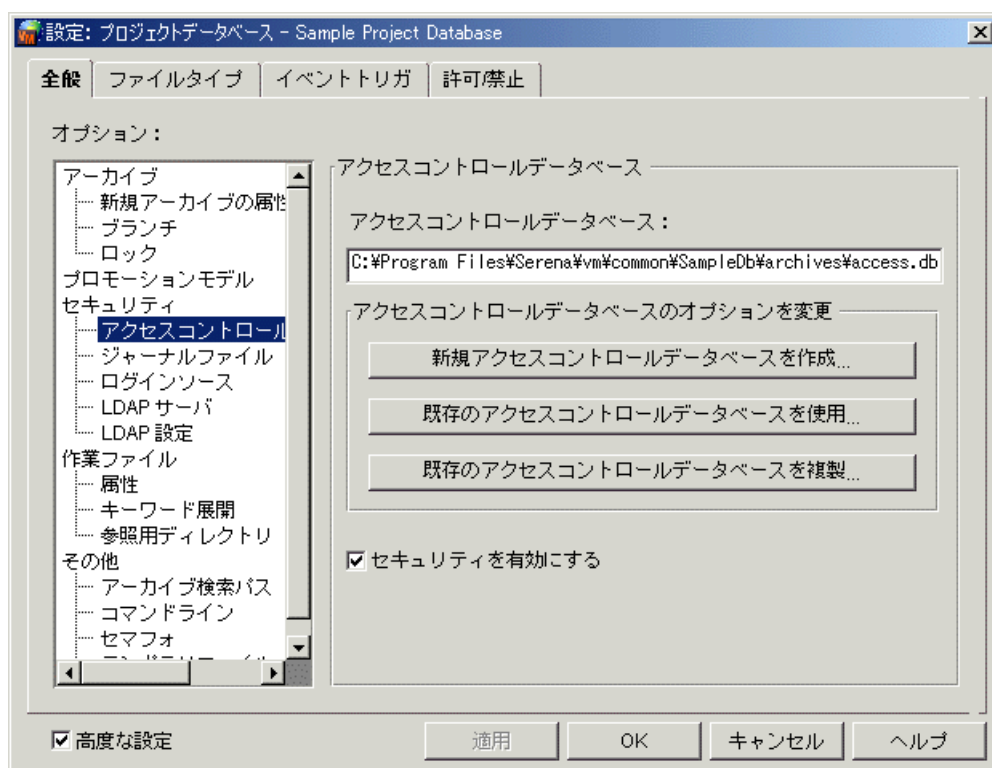
セキュリティの有効化

アクセスコントロールデータベースの定義が終了したら、プロジェクトデータベースまたはプロジェクトのセキュリティコントロールにそれを使用するよう、Version Manager を設定する必要があります。

セキュリティのためにアクセスコントロールデータベースを有効にするには

- 1 このアクセスコントロールデータベースによりセキュリティが制御されるプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。

- 4 <オプション> リストから「セキュリティ」の下の「アクセスコントロールデータベース」を選択します。このアクセスコントロールデータベースに関する情報が右側の<アクセスコントロールデータベース>表示部に表示されます。



- 5 <アクセスコントロールデータベース> テキストボックスに表示された場所と名前を確認し、必要に応じて修正します。
- 6 <アクセスコントロールデータベースのセキュリティを有効にする> チェックボックスを選択します。
- 7 【OK】をクリックします。

プロジェクトに関連付けられたアクセスコントロールデータベース中にユーザ ID が定義されていないためにユーザがアクセスできないプロジェクトは、赤い X が付いたフォルダとしてデスクトップクライアントに表示されます。



コマンドラインインターフェイスを使用する場合

次の基本的な手順は、コマンドラインインターフェイスを使用してセキュリティを設定する際の概要を示したものです。詳細な手順はその次に示します。

- 1 次のものを含むアクセスコントロールテキストファイルを作成します。
 - 使用するカスタム権限セットの定義
 - ユーザの定義と、ユーザへの権限の割り当て
 - アクセスリストグループの定義と、そのグループへの権限とユーザの割り当て

- 2 アクセスコントロールデータベースを作成します。
- 3 使用するアクセスリストを定義します。
- 4 セキュリティを有効にします。

アクセスコントロールテキストファイルの作成

アクセスコントロールデータベースを作成するには、まずアクセスコントロールテキストファイルを作成する必要があります。次にこのテキストファイルを使用してアクセスコントロールデータベースを作成します。

テキストエディタを使用して、アクセスコントロールテキストファイルを作成します。次に 2 つのカスタム権限セット、5 人のユーザ、および 2 つのグループを定義するアクセスコントロールテキストファイルの例を示します。

```
# Define two custom privilege sets
privilege Update: Unlimited, NoDeleteRev, ¥
    NoDeleteVersion
privilege View: Get, ViewArchive
# Define five users
user annab/ (superuser)
user jimp/jimp ()
user kayj/kayj ()
user samg/samg ()
user timf/timf ()
# Define two access list groups
    group PROGRAM (Update): jimp kayj
group DOCUMENT (View): samg timf
```

この例では、最初のコメント行に続く 3 行でカスタム権限セットが定義されます。Administrator である Annab には、SuperUser の権限が与えられています。したがって、このユーザはどのアーカイブに対するどの Version Manager アクションでも行うことができます。ほかの 4 人のユーザは空白のカッコ、() で示されるように Unlimited の権限を与えられています。ただし、この 4 人のユーザのアーカイブへのアクセスは、グループの定義によって制限されます。グループはユーザのアクセスを制限することはできませんが、権限を拡張することはできません。

アクセスコントロールテキストファイルの作成に関しては、次のルールが適用されます。

- それぞれの権限セット、アクセスリストグループ、およびユーザは別々の行に定義します。
- ユーザ、アクセスリストグループ、および権限の名前の長さは、30 文字までに制限されています。
- コメント行の行頭には、シャープ (#) または感嘆符 (!) を付けます。次にその例を示します。
Define two custom privilege sets.
- 行を継続するには円記号 (¥) を使用します。
- ユーザ ID、グループ、および権限では、NoCase ディレクティブを使用しないかぎり、大文字と小文字が区別されます。
- アクセスコントロールテキストファイルについてはどのような名前でも使用できます。

次のセクションでは、カスタム権限セット、ユーザ、およびアクセスリストグループを定義するための構文について説明します。

カスタム権限セットの定義

構文 `privilege [=] custom_name: ¥
component[, component...] ¥
[:promo_group[, promo_group...]]`

ここで、

`custom_name` は定義するカスタム権限セットの名前です。この名前の長さは 30 文字までです。

`component` はカスタム権限セットを構成するベース、コンポジット、または権限セットを指定します。値が複数ある場合はコンマで区切ります。

`promo_group` はこの権限が割り当てられるプロモーショングループを指定します。256 ページの『ユーザによるプロモートの制限』を参照してください。

例 次の例は、Get、Lock、Unlock、Put、および StartBranch のベース権限およびコンポジット権限から成る、Update という名前のカスタム権限セットを定義しています。
`privilege Update: Get, Lock, Unlock, Put, ¥ StartBranch`

ユーザの定義

有効な Version Manager ユーザを特定して権限を割り当てるには、そのユーザ ID をアクセスコントロールデータベース内で指定します。

構文 `user [=] user_id[/password] ¥
[(privilege, privilege...)] ¥
[-d date_range]`

ここで、

`user_id` は Version Manager によるユーザの識別に使用されます。ユーザがアクセスできるようにするには、Version Manager がログインソースから取得した値とこの値を一致させる必要があります。208 ページの『セキュリティの計画』を参照してください。

`password` は、使用されるコンフィグレーションファイルにおいて VLOGIN ログインソースが指定されている場合に、ユーザのパスワードを定義します（デスクトップクライアントのみ）。VLOGIN を Version Manager がユーザを識別するためのソースとして定義していない場合は、パスワードは指定しないでください。

`privilege` は、このユーザに割り当てられるベース権限、コンポジット権限、権限セットを指定します。カスタム権限セットを指定するには、それをあらかじめアクセスコントロールデータベース中で定義しておく必要があります。権限を指定しない場合は、ユーザには Unlimited 権限が与えられます。複数の権限がある場合は、コンマで区切ります。

`date_range` は、ユーザ ID が有効とされる期間です。`date_range` の構文は `ddmmyyyy*ddmmyyyy` で、たとえば `-d 01jan2003*31dec2003` のようになります。

例 次の例ではユーザ ID は `annab` で、パスワードは設定されておらず、このユーザ ID は 2003 年 1 月 1 日から 12 月 31 日まで有効とされます。また `annab` には SuperUser の権限が与えられています。
`user annab/ (superuser)¥
-d 01jan2003*31dec2003`

アクセスリストグループの定義

アクセスリストグループはユーザのセットです。グループのメンバーはプロジェクト内で同一タイプの担当責任を持ち、同一の権限を必要とします。

構文 `group [=] group_name [(privilege, privilege...)] ¥ [:] member, [member...]`

ここで、

`group_name` は定義するアクセスリストグループの名前です。長さは 30 文字以内に制限されています。

`privilege` は、このグループに割り当てられるベース権限、コンボジット権限、権限セットを指定します。カスタム権限セットを指定するには、それをあらかじめアクセスコントロールデータベース中で定義しておく必要があります。権限を指定しない場合には、グループには Unlimited 権限が与えられます。複数の権限がある場合は、コンマで区切ります。

`member` は、ユーザ ID またはあらかじめ定義されたグループ名です。複数のメンバーがある場合はコンマで区切ります。

例 次の例では、ユーザの marym は 3 つのグループに属します。このユーザはグループ、eng1 に属し、このグループに属することにより、グループ、support と engineering のメンバーともなっています。eng1 グループには、空白のかわりに、() で示されるように、Unlimited の権限が与えられています。support グループにはカスタム権限セットの BasicUser が割り当てられています。さらに、engineering グループにはカスタム権限セットの Dev が割り当てられています。marym には、次のような所属するグループの権限の和集合が与えられます。

```
group eng1 (): marym, steveb, marvinp
group support (BasicUser): eng1, marthac, adamj
group engineering (Dev): eng1, toms
```

アクセスコントロールデータベースの作成

アクセスコントロールテキストファイルからアクセスコントロールデータベースを作成するには、次のように `makedb` コマンドを使用します。

```
makedb -adatabase_name text_file
```

ここで、

`database_name` は作成するアクセスコントロールデータベースの名前と場所です。

`text_file` は、アクセスコントロールテキストファイルの名前と場所です。



注 `readdb` コマンドの出力結果に対して `makedb` コマンドを使用する場合は、`makedb` を使用する前にテキストファイルを編集する必要があります。`readdb` により、ユーザはファイルの先頭に置かれるため、ファイルを編集して、ユーザをファイルの末尾に移動する必要があります。`makedb` は、`bin` ディレクトリの下にある `admin` サブディレクトリに置かれています。

次に示す例では、`c:¥pvcs` (UNIX では `/usr/pvcs`) にあるテキストファイル、`access.txt` から、アクセスコントロールデータベース、`access.db` を `c:¥pvcs` (UNIX では `/usr/pvcs`) に作成します。

Windows `makedb -ac:¥pvcs¥access.db c:¥pvcs¥access.txt`

UNIX `makedb -a/usr/pvcs/access.db /usr/pvcs/access.txt`



注 ユーザの有効期限が設定されたアクセスコントロールデータベースに対して `readdb` が使用されると、`makedb` は、`readdb` コマンドの出力結果を読み込むことができません。`makedb` は、`access.txt` ファイル中の、ユーザに設定された有効期限を読み込むことはできません。

アクセスリストの定義

アクセスリストを使用するには、あらかじめアクセスコントロールデータベースを定義しておく必要があります。

新規アーカイブに対してアクセスリストを指定するには、AccessList ディレクティブを使用します。コンフィグレーションファイル中に AccessList ディレクティブが存在する場合は、作成する各アーカイブにこのディレクティブで指定されたアクセスリストが存在します。

既存のアーカイブに対してアクセスリストを指定するには、VCONFIG コマンドと -A パラメータを使用します。このコマンドについては『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

構文 `AccessList = user_id[,user_id...]`

ここで、*user_id* はアーカイブへのアクセスを許可されたユーザまたはユーザグループです。ユーザ ID またはグループ名の大きい文字と小さい文字の使用は、NoCase ディレクティブが使用されていない限り、アクセスコントロールデータベース中での大きい文字と小さい文字の使用と一致する必要があります。

例 この例では、新規に作成されるすべてのアーカイブに対し、2 つのアクセスリストグループを配置します。

```
AccessList = engineering, qa
```

セキュリティの有効化

アクセスコントロールデータベースの定義が終了したら、プロジェクトデータベースまたはプロジェクトのセキュリティコントロールにそれを使用するよう、Version Manager を設定する必要があります。これを行うには、アクセスコントロールデータベースを Version Manager に埋め込むか (227 ページの『Version Manager へのアクセスコントロールデータベースの埋め込み』参照)、そのアクセスコントロールデータベースを (ここで説明するように) コンフィグレーションファイル中で指定します。コンフィグレーションファイル中で指定されたアクセスコントロールデータベースは、コンフィグレーションファイルが次の状態の場合に使用されます。

- Version Manager に埋め込まれている。
- コマンドを発行するときにコマンドラインで指定されている。ほとんどのコマンドでは、コンフィグレーションファイルを指定するための -C コマンドラインオプションを使用できます。
- VCSCFG 環境変数中で定義されている。

アクセスコントロールは、マスターコンフィグレーションファイルで定義する必要があります。

マスターコンフィグレーションファイル中でアクセスコントロールデータベースを指定するには

- 1 テキストエディタでコンフィグレーションファイルを開きます。
- 2 マスターコンフィグレーションファイルに次の行を追加します。

```
AccessDB = database_name  
AccessControl
```

ここで *database_name* は、作成したアクセスコントロールデータベースの名前と場所です。

- 3 オプションとして、ほかのユーザがほかのコンフィグレーションファイルによって設定を上書きできないよう、AccessDB および AccessControl ディレクティブを無効にすることができます。無効にするには、マスターコンフィグレーションファイルに次の行を追加します。

```
Disallow AccessDB  
Disallow AccessControl NoAccessControl
```

- 4 ファイルを保存し、テキストエディタを終了します。

Version Manager へのアクセスコントロールデータベースの埋め込み

アクセスコントロールデータベースを Version Manager に埋め込むことにより、すべてのユーザが同一のセキュリティ定義を使用し、またほかのアクセスコントロールデータベースを使用できないことが保証されます。Version Manager に埋め込まれたアクセスコントロールデータベースは、その Version Manager を使用するすべてのデスクトップクライアントおよびコマンドラインインターフェイスユーザに影響します。



重要！ Version Manager に埋め込まれたアクセスコントロールデータベースは、有効に設定されないかぎり機能しません。デフォルトではアクセスコントロールデータベースは有効になっていません。

デスクトップクライアントを使用する場合

アクセスコントロールデータベースを埋め込むには

- 1 Version Manager に埋め込む対象であり、アクセスコントロールデータベースに関連付けられたプロジェクトデータベースを選択します。
- 2 [管理] - [設定の保護] を選択します。[設定の保護] ダイアログボックスが表示されます。

- 3 〈アクセスデータベース〉チェックボックスを選択します。このチェックボックスの隣のテキストボックスには、埋め込もうとするアクセスコントロールデータベースの場所と名前が表示されます。このフィールドは変更できません。
- 4 【OK】をクリックします。

コマンドラインインターフェイスを使用する場合

アクセスコントロールデータベースを埋め込むには、**VCONFIG** コマンドを次のように使用します。

構文 `vconfig -adatabase_name vm_filename`

ここで、

`database_name` は、アクセスコントロールデータベースの名前と場所です。

`vm_filename` は次のいずれかになります。

- Windows : vmwfcv.dll
- UNIX : vmufvc.a



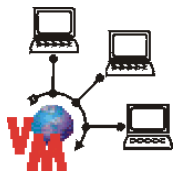
注 vconfig は、bin ディレクトリのサブディレクトリ、admin にあります。

アクセスコントロールデータベースの名前を埋め込んだ後、ファイル (vmwfcv.dll または vmufvc.a) を Version Manager の実行ファイルと同じ場所に移動する必要があります (まだ移動されていない場合)。また、ユーザがこの場所に対して書き込み許可を持っていないことを確認します。これは、ユーザが Version Manager に別のアクセスコントロールデータベースを埋め込むことを防止するためです。

VCONFIG コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

プロジェクトデータベース作成の制限

ユーザによるプロジェクトデータベースの作成を制限するには、作成する前にユーザが入力しなければならないパスワードを定義する必要があります。プロジェクトデータベースの作成を許可しないユーザには、このパスワードを知らせないようにします。



注 次のセクションは、ファイルサーバ以外のプロジェクトデータベースにのみ適用されます。ファイルサーバのプロジェクトデータベースに対しては、パスワードは Version Manager File Server の管理ユーティリティで設定します。141 ページの『[パスマップセキュリティオプションの設定](#)』を参照してください。

定義したパスワードは、Windows 版では vmwfcvj.dll、UNIX 版では vmufvcj.a というセキュリティ設定ファイルに埋め込まれます。このファイルは、Windows 版では `install_dir\vm\common\bin\win32`、UNIX 版では `install_dir/vm/common/bin/OS_Name` というディレクトリにインストールされます。デフォルトでは、このファイルにはパスワードは埋め込まれていません。したがって、どのユーザもプロジェクトデータベースを作成できます。

このファイルにパスワードを定義すると、このファイルを含む Version Manager を使用するデスクトップクライアントおよびプロジェクトコマンドラインインターフェイスのユーザすべてに影響します。パスワードを定義する管理者は、bin ディレクトリへの書き込み権限を持っている必要があります。ほかのユーザには、パスワードを定義することも、変更することもできないよう、このディレクトリ、またはセキュリティ設定ファイルへの書き込み権限を与えないようにしてください。

ほかのユーザは、管理者がプロジェクトデータベースの作成が制限されるよう設定した、同じ Version Manager を使用する必要があります。このためには、各ユーザが Version Manager の

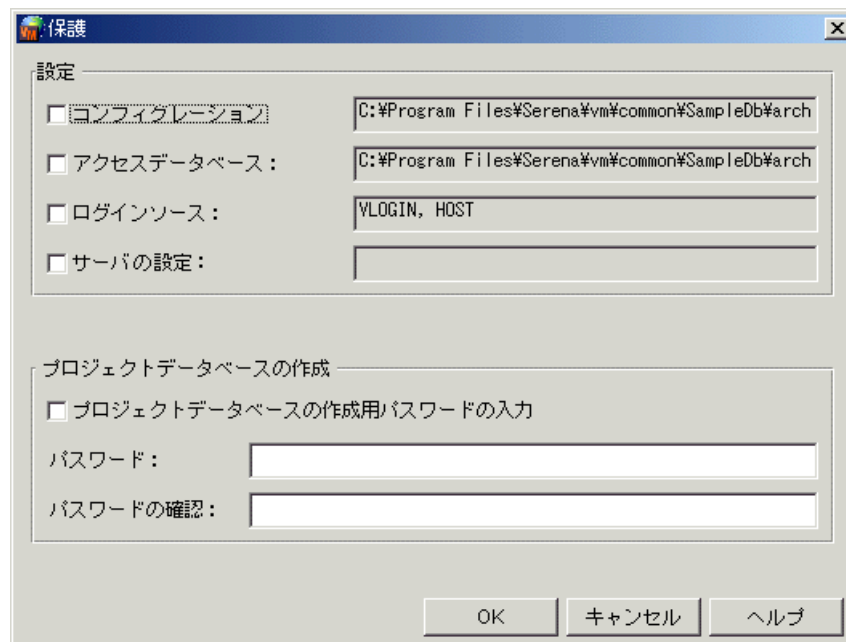
ワークステーションインストールを実行しなければなりません。ワークステーションインストールについて詳しくは、『Serena ChangeMan Version Manager インストールガイド』を参照してください。

プロジェクトデータベースの作成がパスワードにより制限されている場合は、ユーザが [管理] - [プロジェクトデータベースの作成] を選択するか、またはプロジェクトデータベースのどれかを選択して [編集] - [コピー] を選択すると、デスクトップクライアントによりパスワードの入力が求められます。このパスワードは、管理者が定義したものと完全に同じものにし、アルファベットの大文字と小文字を区別して入力しなければいけません。

パスワードを入力しないとプロジェクトデータベースの作成機能を使用できない場合は、Version Manager を初めて起動したとき、[Serena ChangeMan Version Manager へようこそ] ダイアログボックスの〈新規プロジェクトデータベースを作成する〉チェックボックスはユーザに対して表示されません。

プロジェクトデータベースの作成を制限するには

- 1 [管理] - [設定の保護] を選択します。[設定の保護] ダイアログボックスが表示されます。



- 2 〈プロジェクトデータベースの作成用パスワードの入力〉チェックボックスを選択します。
- 3 〈パスワード〉フィールドにパスワードを半角英数字で指定します。パスワードでは、アルファベットの大文字と小文字が区別されます。ここに入力した文字はすべて、アスタリスク (*) で表示されます。
- 4 〈パスワードの確認〉フィールドに、パスワードを再度入力します。

セキュリティのメンテナンス

セキュリティの設定後に、退職、スタッフの増員、プロジェクトの変更などに伴い、変更を行う必要が発生します。このセクションでは、セキュリティをメンテナンスするための手順について説明します。

アクセスコントロールデータベースの設定を表示するには、[管理] - [セキュリティ] - [ユーザ] を選択します。302 ページの『セキュリティレポートの生成』を参照してください。

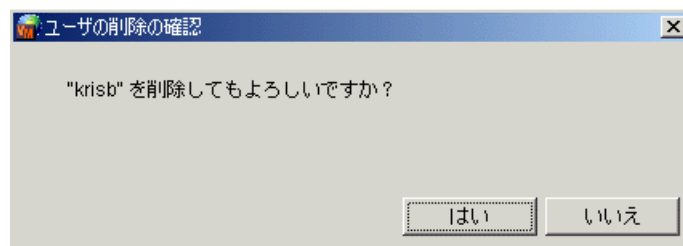
アクセスコントロールデータベースからのユーザの削除

アクセスコントロールデータベースからユーザを削除するには

- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [ユーザ] を選択します。[プロジェクトセキュリティ] ダイアログボックスが表示されます。



- 3 <すべてのユーザ> リストから削除するユーザを選択し、[削除] アイコンをクリックします。確認のためのダイアログボックスが表示されます。



- 4 選択したユーザを削除するには、【はい】をクリックします。

ユーザの ID の変更

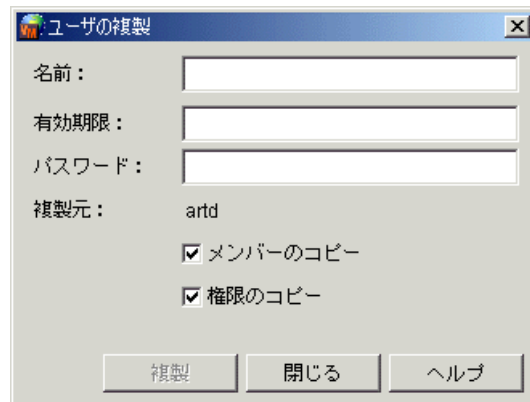
ユーザ ID を変更するには、ユーザを複製し、複製した（新しい）ユーザ定義中のユーザ ID を変更し、次に古いユーザ定義を削除する必要があります。

ユーザの ID を変更するには


- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [ユーザ] を選択します。[プロジェクトセキュリティ] ダイアログボックスが表示されます。

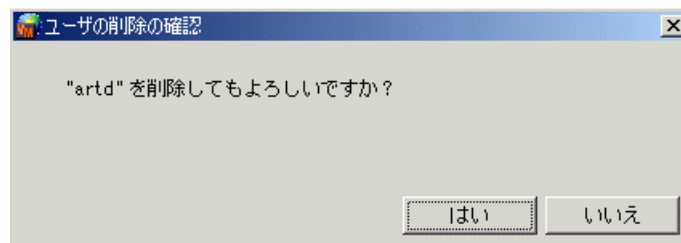


- 3 <すべてのユーザ> リストから削除するユーザを選択し、[複製] アイコンをクリックします。[ユーザの複製] ダイアログボックスが表示されます。



〔ユーザーの複製〕ダイアログボックス内で次の作業を行います。

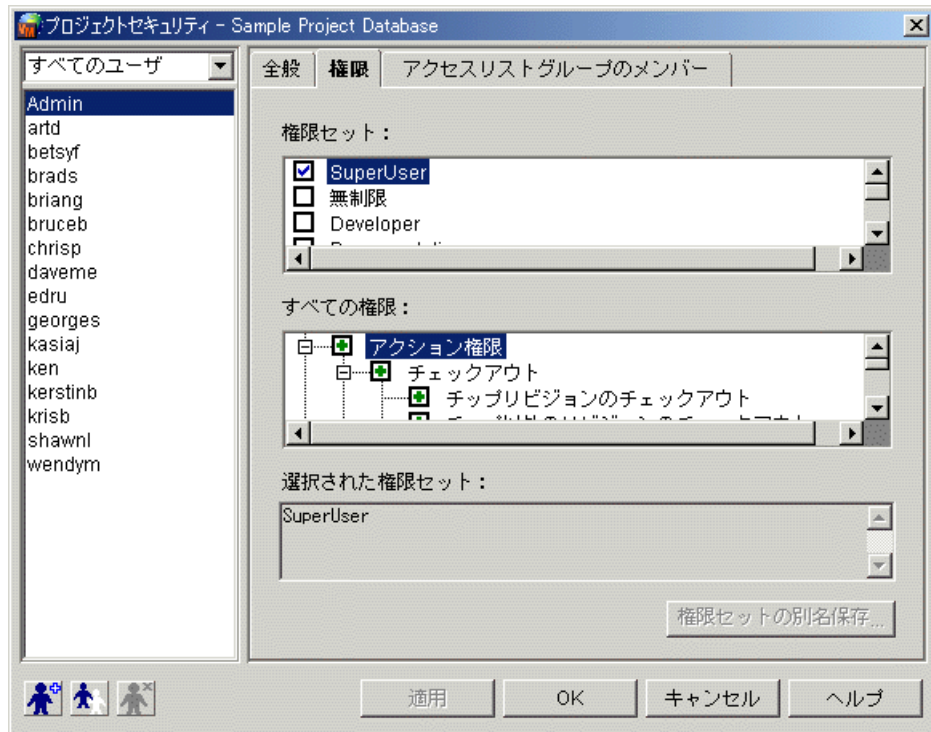
- a <名前> フィールドのユーザ ID を変更します。
- b 【閉じる】をクリックします。〔ユーザーの複製〕ダイアログボックスが閉じ、〔プロジェクトセキュリティ〕ダイアログボックスが表示されます。
- 4 〔プロジェクトセキュリティ〕ダイアログボックス内で、〈すべてのユーザ〉リストから古いユーザ定義を選択します。
- 5  [削除] アイコンをクリックします。確認のためのダイアログボックスが表示されます。



- 6 選択したユーザを削除するには、【はい】をクリックします。

ユーザ権限の変更

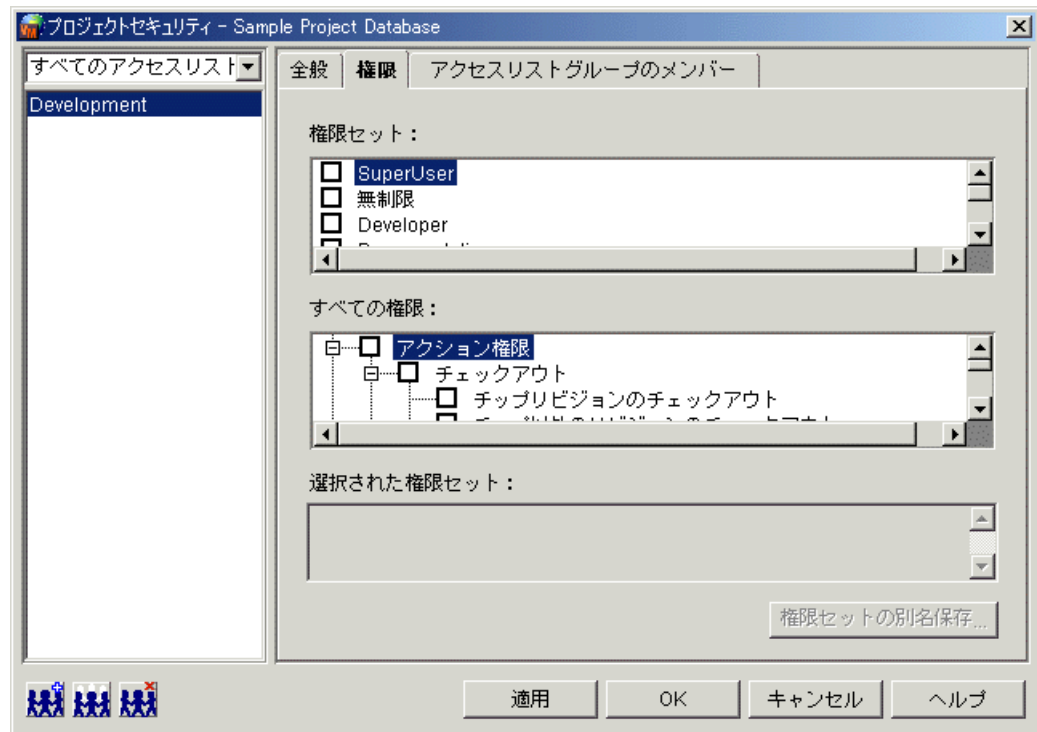
- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [ユーザ] を選択します。〔プロジェクトセキュリティ〕ダイアログボックスが表示されます。
- 3 〈すべてのユーザ〉リストから削除するユーザを選択し、[複製] アイコンをクリックします。〔ユーザーの複製〕ダイアログボックスが表示されます。
- 4 [権限] タブをクリックします。このタブ上でユーザに割り当てる権限および権限セットを選択します。



- 5 【適用】をクリックします。
- 6 さらにほかのユーザの権限を変更するには手順 3 ~ 5 を繰り返します。変更しない場合は、【OK】をクリックして手順を終了します。

アクセスリストグループ権限の変更

- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [アクセスリストグループ] を選択します。[プロジェクトセキュリティ] ダイアログボックスが表示されます。
- 3 <すべてのアクセスリスト> リストから権限を変更するグループを選択します。
- 4 [権限] タブをクリックします。このタブ上でグループに割り当てる権限および権限セットを選択します。



- 5 【適用】をクリックします。
- 6 さらにほかのグループの権限を変更するには手順 3～5 を繰り返します。変更しない場合は、【OK】をクリックして手順を終了します。

アクセスリストグループのメンバーの変更


- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [アクセスリストグループ] を選択します。[プロジェクトセキュリティ] ダイアログボックスが表示されます。
- 3 <すべてのアクセスリスト> リストからメンバーを変更するグループを選択します。

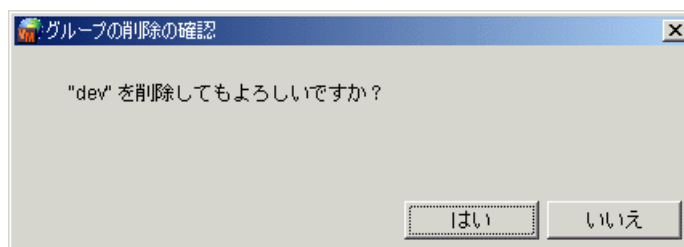
- 4 [アクセスリストグループのメンバー] タブをクリックします。このタブ上でグループに割り当てるユーザを選択します。



- 5 【適用】をクリックします。
- 6 さらにほかのグループのメンバーを変更するには手順 3 ~ 5 を繰り返します。変更しない場合は、【OK】をクリックして手順を終了します。

アクセスコントロールデータベースからのアクセスリストグループの削除

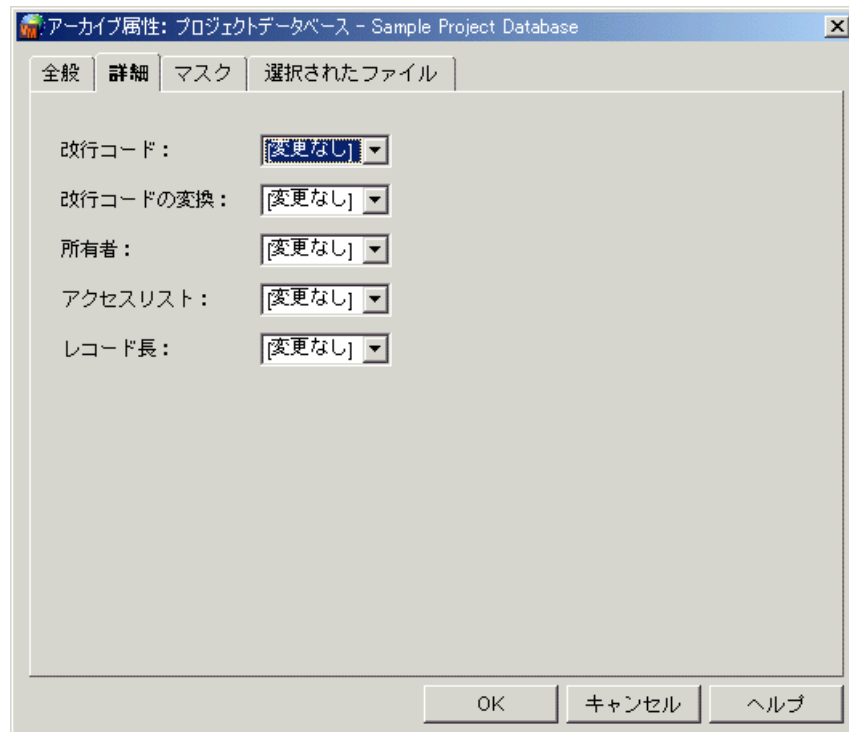
- 1 変更するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [アクセスリストグループ] を選択します。[プロジェクトセキュリティ] ダイアログボックスが表示されます。
- 3  <すべてのアクセスリスト> リストから削除するグループを選択し、[削除] アイコンをクリックします。確認のためのダイアログボックスが表示されます。



- 4 選択したグループを削除するには【はい】をクリックします。

アクセスリストの変更

- 1 変更するアクセスリストに関連付けられた、プロジェクトまたはバージョン管理ファイルを選択します。プロジェクトを選択した場合は、そのオブジェクトのすべてのアーカイブについて、アクセスリストが変更されます。
- 2 [管理] - [アーカイブ属性] を選択します。[アーカイブ属性] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 [詳細設定] タブをクリックします。



- 4 <アクセスリスト> リストから「削除」、「変更」、または「追加」を選択します。「変更」を選択した場合は、アーカイブにアクセスできるアクセスリストグループと個別のユーザを、右側に表示されるテキストボックスに入力します。リスト中の各項目はコンマ (,) で区切ります。あるいは【...】をクリックして、グループまたは個別のユーザを選択します。このフィールドには、合計で 254 文字までの値を入力できます。

「追加」を選択した場合は、<アクセスリスト> テキストボックスの右側に表示されるテキストボックスに追加のアクセスリストグループとユーザを入力します。

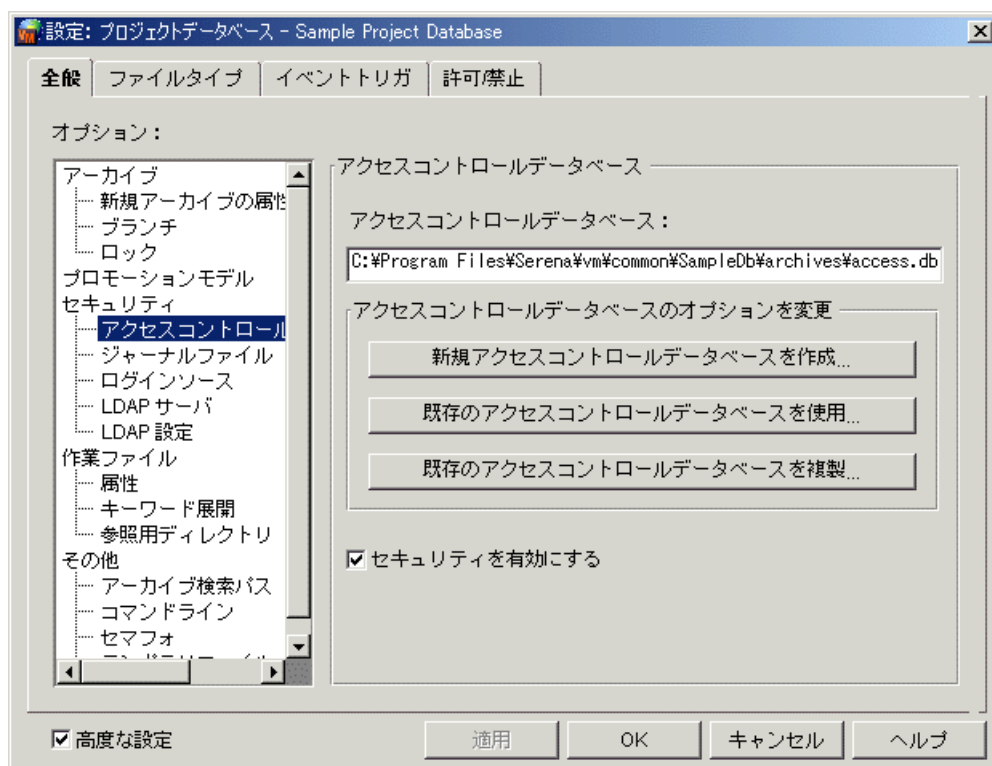
ここで指定するアクセスリストグループとユーザは、アクセスコントロールデータベース中で定義されている必要があります。

- 5 [OK] をクリックします。

セキュリティの無効化

- 1 無効にするアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。

- 3 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。
- 4 〈オプション〉リストから「セキュリティ」の下の「アクセスコントロールデータベース」を選択します。このアクセスコントロールデータベースに関する情報が右側の〈アクセスコントロールデータベース〉表示部に表示されます。

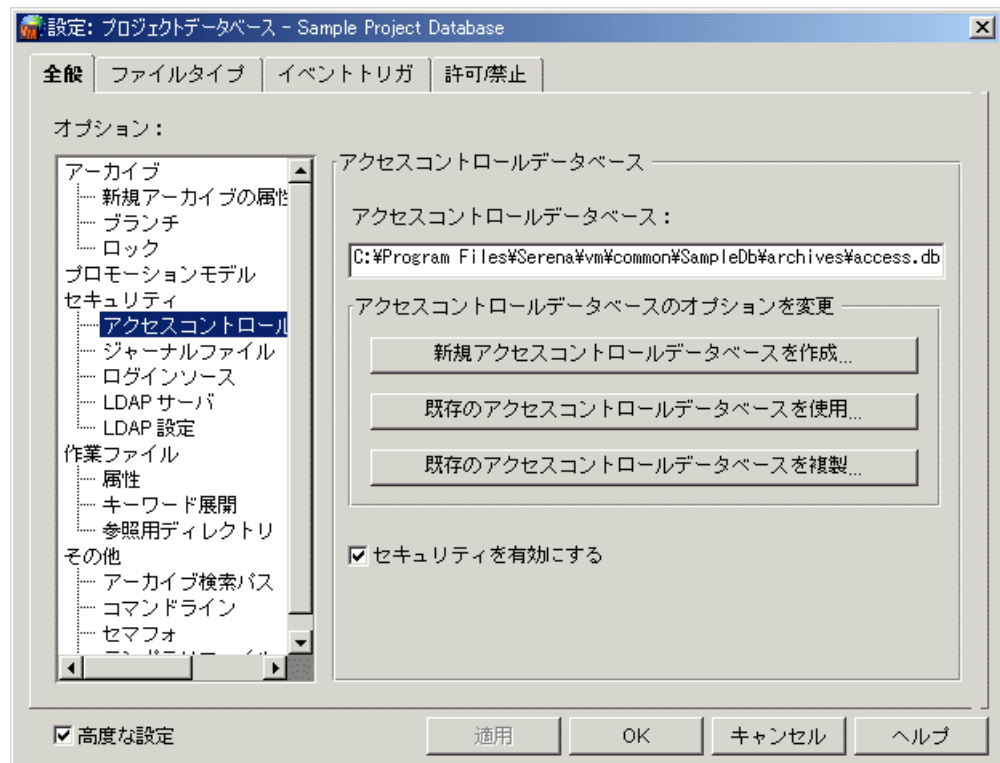


- 5 〈セキュリティを有効にする〉チェックボックスの選択を解除します。
- 6 【OK】をクリックします。これで、このプロジェクトデータベースまたはプロジェクトが、アクセスコントロールデータベースによるセキュリティ管理の対象ではなくなります。

プロジェクトデータベースに関連付けられたアクセスコントロールデータベースの変更

- 1 アクセスコントロールデータベースを変更するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 〈高度な設定〉チェックボックスがまだ選択されていない場合は選択します。

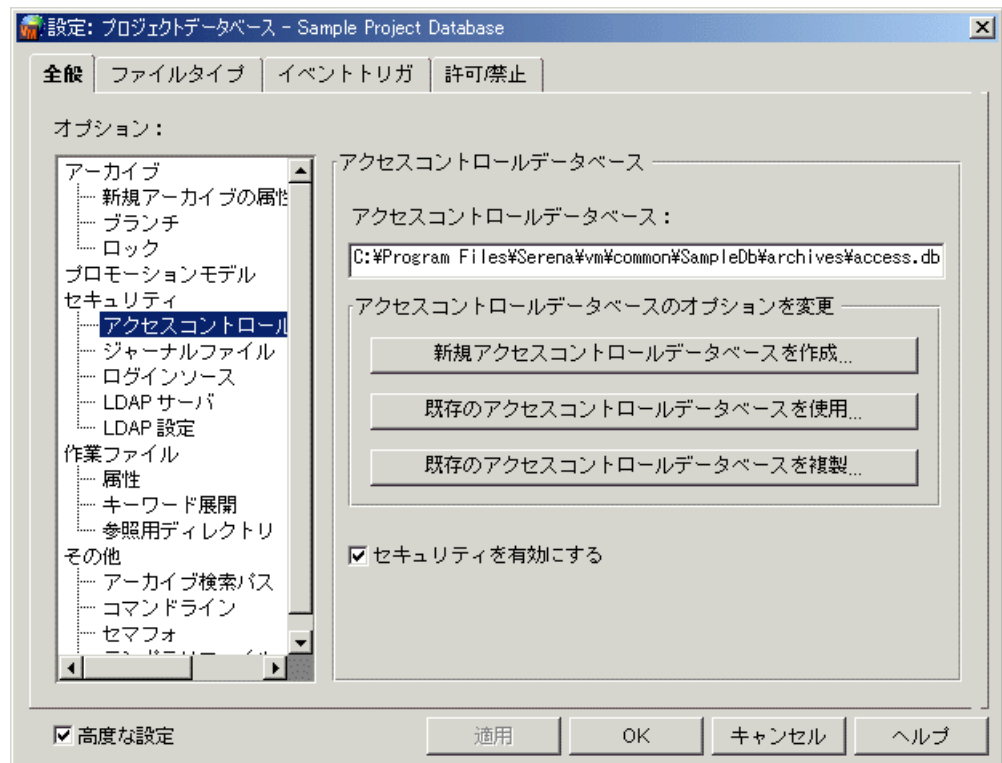
- 4 <オプション> リストから「セキュリティ」の下の「アクセスコントロールデータベース」を選択します。このアクセスコントロールデータベースに関する情報が右側の<アクセスコントロールデータベース>表示部に表示されます。



- 5 <アクセスコントロールデータベース> テキストボックスには、プロジェクトデータベースまたはプロジェクトに現在関連付けられているアクセスコントロールデータベースが表示されます。ほかのアクセスコントロールデータベースを使用するには、次のいずれかの作業を行います。
- 【新規アクセスコントロールデータベースを作成】をクリックし、プロジェクトデータベースまたはプロジェクトのアーカイブディレクトリに、新しいアクセスコントロールデータベースを作成する。デフォルトのアクセスコントロールデータベース (default.db) がこの場所にコピーされます。
 - 既存のアクセスコントロールデータベースを使用するには、【既存のアクセスコントロールデータベースを使用】をクリックする。[アクセスコントロールデータベース] ダイアログボックスが表示され、アクセスコントロールデータベースを選択することができます。このオプションは、同一のセキュリティ情報を使用するプロジェクトデータベースを、複数作成する場合に便利です。それぞれのプロジェクトデータベースについて同一のアクセスコントロールデータベースを使用している場合、セキュリティの必要が変化するとき、たとえばアクセスコントロールデータベースに新規ユーザを追加するときなどには、1つのアクセスコントロールデータベースを更新するだけで済みます。
 - 既存のアクセスコントロールデータベースを複製するには、【既存のアクセスコントロールデータベースをコピー】をクリックする。アクセスコントロールデータベースを選択するため [アクセスコントロールデータベースの選択] ダイアログボックスが表示されます。Version Manager は選択したアクセスコントロールデータベースのコピーを作成し、それをプロジェクトデータベースのアーカイブディレクトリに配置します。
- 6 【OK】をクリックします。

アクセスコントロールデータベースとプロジェクトデータベースとの関連付けの削除

- 1 アクセスコントロールデータベースを削除するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 4 <オプション> リストから「セキュリティ」の下の「アクセスコントロール」を選択します。このアクセスコントロールデータベースに関する情報が右側の <アクセスコントロールデータベース> 表示部に表示されます。



- 5 <アクセスコントロールデータベース> グループボックスには、現在プロジェクトデータベースまたはプロジェクトに関連付けられているアクセスコントロールデータベースが表示されます。このアクセスコントロールデータベースとの関連付けを削除するには、<アクセスコントロールデータベース> テキストボックスの値を削除して空白にします。
- 6 [OK] をクリックします。

権限の定義

このセクションでは、ベース権限、コンポジット権限、およびデフォルトの権限セットを、3つの表に示します。

表 6-1 には、ベース権限のリスト、それらによって可能となるアクション、およびそれらによって使用可能となるデスクトップクライアントコマンドまたはダイアログボックスフィールドを示します。権限を有効にするコマンドラインインターフェイスのコマンドについては、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

表 6-1 : ベース権限

コマンドラインでのベース権限名 デスクトップクライアントでのベース権限名	アクション、使用できるデスクトップクライアントコマンドまたはダイアログボックスのフィールド
アクションの権限	
GetTip チップリビジョンのチェックアウト	チップリビジョンをチェックアウトします。 Merge Tool を起動します。 相違点レポートを作成します。 チップリビジョンを選択したときの [アクション] - [チェックアウト]、[アクション] - [マージの表示]、および [アクション] - [相違点の表示]。
GetNonTip チップ以外のリビジョンのチェックアウト	チップ以外のリビジョンをチェックアウトします。 Merge Tool を起動します。 相違点レポートを作成します。 チップ以外のリビジョンを選択したときの [アクション] - [チェックアウト]、[アクション] - [マージの表示]、および [アクション] - [相違点の表示]。
PutTrunk トランクへのチェックイン	トランクリビジョンを保存します。 チップリビジョンを選択したときの [アクション] - [チェックイン]。 [ファイル] - [作業ファイルの追加]。
PutBranch ブランチへのチェックイン	ブランチリビジョンを保存します。 チップ以外のリビジョンを選択したときの [アクション] - [チェックイン]。
StartBranch ブランチの作成	ブランチを作成します。 [チェックイン] ダイアログボックスの〈強制的にブランチさせる〉チェックボックスを使用可能にします。
LockTip チップリビジョンのロック	チップリビジョンをロックします。 チップリビジョンを選択したときの [アクション] - [リビジョンのロック]。
LockNonTip チップ以外のリビジョンのロック	チップ以外のリビジョンをロックします。 チップ以外のリビジョンを選択したときの [アクション] - [リビジョンのロック]。
Unlock ロック解除	ユーザが所有するリビジョン上のロックを解除します。 この権限と「ほかのユーザのロック解除」(Break Lock) 権限の両方が付与された場合 : [アクション] - [リビジョンのロック解除]。
BreakLock ほかのユーザのロック解除	リビジョンのロックを解除します。 [リビジョンのロック解除] ダイアログボックスの〈ユーザ〉フィールドを使用可能にします。
AddVersion バージョンラベルの割り当て	リビジョンにバージョンラベルを割り当てます。 [アクション] - [バージョンラベル] - [割り当て] (DeleteVersion と組み合わせてバージョンラベルを変更する : [アクション] - [バージョンラベル] - [ファイル名の変更])
ModifyVersion バージョンラベルの変更	別のリビジョンにバージョンラベルを割り当てます。 [アクション] - [バージョンラベル] - [割り当て]

コマンドラインでのベース権限名 デスクトップクライアントでのベース権限名	アクション、使用できるデスクトップクライアントコマンドまたはダイアログボックスのフィールド
DeleteVersion バージョンラベルの削除	バージョンラベルを削除します。 [アクション] - [バージョンラベル] - [削除] (AddVersion と組み合わせてバージョンラベルを変更する : [アクション] - [バージョンラベル] - [ファイル名の変更])
AddGroup プロモーショングループの割り当て	リビジョンにプロモーショングループを割り当てます。 [アクション] - [プロモーショングループ] - [割り当て]
ModifyGroup プロモーショングループの変更	プロモーショングループの名前を変更 / 移動します。 [アクション] - [プロモーショングループ] - [変更]
DeleteGroup プロモーショングループの削除	リビジョンからのプロモーショングループの関連付けを解除します。 [アクション] - [プロモーショングループ] - [削除]
Promote 次のグループへプロモート	リビジョンをプロモートします。 [アクション] - [プロモーショングループ] - [プロモート]
ViewAccessDB アクセスコントロールデータベースの表示	セキュリティレポートを作成します。 [管理] - [セキュリティ] - [レポートの表示]
アーカイブ権限	
InitArchive アーカイブと作業ファイルの作成	アーカイブを作成します。 [ファイル] - [作業ファイルの追加]
ChangeAccessList アーカイブ中のアクセスリストの変更	アクセスリストを修正します。 [アーカイブ属性] ダイアログボックスの〈アクセスリスト〉フィールドを使用可能にします。
ChangeCommentDelimiter コメント接頭辞の変更	コメント接頭辞を変更します。 [アーカイブ属性] ダイアログボックスの〈コメント接頭辞〉フィールドを使用可能にします。
ChangeOwner 所有者の変更	アーカイブの所有者を変更します。 [アーカイブ属性] ダイアログボックスの〈所有者〉フィールドを使用可能にします。
ChangeProtection アーカイブ属性の変更	次のアーカイブ属性を変更します : リビジョンのロック設定、複数リビジョンの同時ロック、キーワードの拡張、UNIX 上で実行される変換、アーカイブの書き込み保護、情報の圧縮、デルタレコードの生成。 [アーカイブ属性] ダイアログボックスの〈保護〉フィールドを使用可能にします。
ChangeWorkfileName バージョン管理ファイル名の変更	アーカイブの作業ファイル名属性を変更します。[アーカイブ属性] ダイアログボックスの〈作業ファイル名〉フィールドを使用可能にします。 バージョン管理ファイル名の変更権限は現バージョンでは実装されていないため、バージョン管理ファイル名は変更できません。
ModifyChangeDescription リビジョンのコメントの変更	リビジョンの変更説明を編集します。 [プロパティ] ダイアログボックスの [リビジョン] タブの〈コメント〉フィールドを使用可能にします。
ModifyWorkfileDescription 作業ファイルのコメントの変更	作業ファイルの説明を変更します。 [プロパティ] ダイアログボックスの [バージョン管理ファイル] タブの〈コメント〉フィールドを使用可能にします。
DeleteRevTip チップリビジョンの削除	チップリビジョンを削除します。 チップリビジョンを選択したときの [アクション] - [リビジョンの削除] - [削除]。

コマンドラインでのベース権限名 デスクトップクライアントでのベース権限名	アクション、使用できるデスクトップクライアントコマンドまたはダイアログボックスのフィールド
DeleteRevNonTip チップ以外のリビジョンの削除	チップ以外のリビジョンを削除します。 チップ以外のリビジョンを選択したときの [アクション] - [リビジョンの削除] - [削除]。
ViewArchiveHeader アーカイブヘッダの表示	アーカイブヘッダ情報を表示します。 [プロパティ] ダイアログボックスの [バージョン管理ファイル] タブに、アーカイブヘッダ情報が表示されます。
ViewArchiveRev アーカイブリビジョンの表示	リビジョン履歴を表示します。 [プロパティ] ダイアログボックスの [バージョン管理ファイル] タブに、バージョン履歴情報が表示されます。
プロジェクト権限	
NoProjectNewProject プロジェクトの作成	プロジェクトを作成します。 [ファイル] - [プロジェクトの作成]
NoFolderChangeFolder プロジェクトの変更	プロジェクトの名称変更とプロジェクトの属性変更を行います。 [ファイル] - [プロパティ] [ファイル] - [ファイル名の変更] ルートワークスペースとパブリックワークスペースの作業ファイルの場所を変更します。
NoProjectDeleteProject プロジェクトの削除	プロジェクトを削除します。 プロジェクトを選択したときの [ファイル] - [削除]。
NoProjectCopyProject プロジェクトのコピー	プロジェクトをコピーします。 プロジェクトを選択したときの [ファイル] - [コピー]。
NoFolderChangeFolderMembers バージョン管理ファイルの追加または削除	プロジェクトデータベースまたはプロジェクトに作業ファイルを追加します。 [ファイル] - [作業ファイルの追加] アーカイブをインポートします。 [管理] - [アーカイブのインポート] プロジェクトデータベースまたはプロジェクトからのバージョン管理ファイルを削除します。 バージョン管理ファイルを選択したときの [ファイル] - [削除]。
NoProjectConfigureProject プロジェクトの設定	プロジェクトまたはプロジェクトデータベースを設定します。 [管理] - [プロジェクトの設定] ルートワークスペースとパブリックワークスペースのベース、ブランチ、およびデフォルトの各プロモーショングループの設定を変更します。
NoOptionsSecurity セキュリティの設定	セキュリティに関するユーザ、グループ、権限、およびログインソースを定義します。 [管理] - [セキュリティ]
NoActionsJournalReport ジャーナルの表示	ジャーナルレポートを作成します。 [アクション] - [ジャーナルの表示]
フォルダ権限 (バージョン 5.3/6.0 プロジェクトのみ。これらの権限はデスクトップクライアントには使用できません。)	
NoFolderNewFolder	フォルダを作成します。 [ファイル] - [フォルダの作成]
NoFolderChangeFolder	フォルダの名前を変更し、フォルダの属性を変更します。 [ファイル] - [プロパティ] [ファイル] - [ファイル名の変更]

コマンドラインでのベース権限名 デスクトップクライアントでのベース権限名	アクション、使用できるデスクトップクライアントコマンドまたはダイアログボックスのフィールド
NoFolderDeleteFolder	フォルダを削除します。 フォルダを選択したときの [ファイル] - [削除]。
NoFolderCopyFolderMembers	フォルダをコピーします。 フォルダを選択したときの [ファイル] - [コピー]。
NoFolderChangeFolderMembers	5.3/6.0 プロジェクトに作業ファイルを追加します。 [ファイル] - [作業ファイルの追加] 5.3/6.0 プロジェクトからバージョン管理ファイルを削除します。 バージョン管理ファイルを選択したときの [ファイル] - [削除]。
NoFolderUpdateProjectFolder	5.3/6.0 プロジェクトを更新します。 [ファイル] - [プロジェクトフォルダの更新]

コンポジット権限

表 6-2 に、コンポジット権限のリストとそれを構成するベース権限を示します。この表ではまずコマンドラインインターフェイスでの名称をリストアップし、次にそれぞれの権限に対するデスクトップクライアントでの名称を示します。各ベース権限の定義については、[239 ページの表 6-1 : ベース権限](#)を参照してください。

表 6-2 : コンポジット権限

コマンドラインでのコンポジット権限名 デスクトップクライアントでのコンポジット権限名	対応するベース権限
アクションのコンポジット権限	
Get チェックアウト	GetNonTip GetTip チップリビジョンのチェックアウト チップ以外のリビジョンのチェックアウト
Put チェックイン	PutBranch PutTrunk チップリビジョンのチェックイン ブランチのチェックイン
Lock ロック	LockTip LockNonTip チップリビジョンのロック チップ以外のリビジョンのロック
バージョンラベル (デスクトップクライアントのみ)	バージョンラベルの割り当て バージョンラベルの変更 バージョンラベルの削除
プロモーショングループ (デスクトップクライアントのみ)	プロモーショングループの割り当て プロモーショングループの変更 プロモーショングループの削除 次のグループへのプロモート
アーカイブのコンポジット権限	

コマンドラインでのコンポジット権限名 デスクトップクライアントでのコンポジット権限名 アーカイブプロパティの変更（デスクトップクライアントのみ）	対応するベース権限 アーカイブアクセスリストの変更 コメント接頭辞の変更 アーカイブ所有者の変更 アーカイブ属性の変更 バージョン管理ファイル名の変更
ModifyDescription コメントの変更	ModifyChangeDescription ModifyWorkfileDescription リビジョンのコメントの変更 作業ファイルのコメントの変更
DeleteRev リビジョンの削除	DeleteRevTip DeleteRevNonTip チップリビジョンの削除 チップ以外のリビジョンの削除
ViewArchive アーカイブ詳細の表示	ViewArchiveHeader ViewArchiveRev アーカイブヘッダの表示 リビジョンの表示
プロジェクトのコンポジット権限	
プロジェクト（デスクトップクライアントのみ）	プロジェクトの作成 プロジェクトの変更 プロジェクトの削除 プロジェクトのコピー 作業ファイルの追加 / 削除

デフォルトの権限セット

表 6-3 に、Version Manager のデフォルトの権限セット、およびデフォルトの権限セットを定義するベース権限、コンポジット権限、およびその他の権限セットを示します。



注 SuperUser および Unlimited 権限セットは変更できません。これら以外の権限セットは、サンプルとして表示されているもので、変更することができます。

表 6-3 : デフォルト権限セット

権限セット	ベース権限
SuperUser	表 6-1 に含まれるすべての権限。さらに、プロジェクトデータベースのコピー / 名前の変更、チェックインを伴わない作業ファイルの追加の各アクション。この権限は、アクセスリストによって制限されません。
Unlimited	表 6-1 に含まれるすべての権限。さらに、チェックインを伴わない作業ファイルの追加のアクション。
開発者	「プロジェクトリーダー」権限セットから、「プロジェクトの作成」、「プロジェクトの設定」、「プロジェクトのコピー」、「プロジェクトの削除」、「バージョン管理ファイルの追加」、「バージョン管理ファイルの削除」、「プロジェクトの変更」、「セキュリティの設定」を除いたもの。

権限セット	ベース権限
プロジェクトリーダー	<p>「チップリビジョンのロック」、「チップ以外のリビジョンのロック」、「ロック解除」、「チップリビジョンのチェックアウト」、「チップ以外のリビジョンのチェックアウト」、「トランクへのチェックイン」、「ブランチへのチェックイン」、「ブランチの作成」、「アーカイブアクセスリストの変更」、「アーカイブ所有者の変更」、「アーカイブ属性の変更」、「コメント接頭辞の変更」、「バージョン管理ファイル名の変更」、「作業ファイルのコメントの変更」、「リビジョンのコメントの変更」、「バージョンラベルの割り当て」、「バージョンラベルの削除」、「バージョンラベルの変更」、「アーカイブの作成」、「チップリビジョンの削除」、「チップ以外のリビジョンの削除」、「アーカイブヘッダの表示」、「リビジョンの表示」、「次のグループへのプロモート」、「プロモーショングループの割り当て」、「プロモーショングループの変更」、「プロモーショングループの削除」、「プロジェクトの作成」、「プロジェクトの設定」、「プロジェクトのコピー」、「プロジェクトの削除」、「バージョン管理ファイルの追加」、「バージョン管理ファイルの削除」、「セキュリティの設定」、および「ジャーナルの表示」。</p>
サポート	<p>「アーカイブヘッダの表示」、「リビジョンの表示」、および「ジャーナルの表示」。</p>
品質保証	<p>「サポート」権限セットおよび「バージョンラベルの割り当て」、「バージョンラベルの削除」、「バージョンラベルの変更」、「次のグループへのプロモート」、「プロモーショングループの割り当て」、「プロモーショングループの変更」、「プロモーショングループの削除」。</p>
ドキュメント	<p>「品質保証」権限セットおよび「チップリビジョンのロック」、「チップ以外のリビジョンのロック」、「ロック解除」、「チップリビジョンのチェックアウト」、「チップ以外のリビジョンのチェックアウト」、「トランクへのチェックイン」、「ブランチへのチェックイン」、「アーカイブの作成」。</p>

第7章

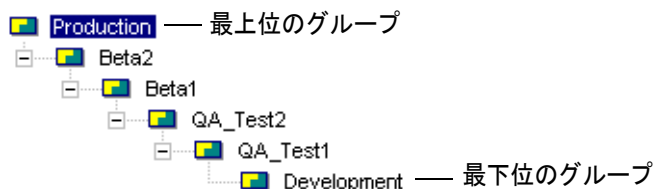
プロモーションモデルの使用

はじめに	246
プロモーションモデルのルール	246
プロモーションモデルの定義	248
プロモーションとライフサイクルの管理	251
プロモーションと並行開発	252
ユーザによるプロモートの制限	256

はじめに

プロモーションモデルとは、開発サイクルにおける到達目標の階層のことです。プロモーションモデルは、設計段階から最終的な出荷に至るまでの、ソースコード開発の管理に使用します。それぞれの到達目標は、プロモーショングループによって表されます。プロモーショングループとしては、Development（開発）、QA（品質管理）、Production（製造）などがあります。

プロモーションモデルの例を次の図に示します。



Serena ChangeMan Version Manager のプロモーションモデルは、ファイルのリビジョンと、開発サイクルにおける到達目標の間の論理的な関連付けをベースとしています。メインフレーム環境におけるプロモーションとは異なり、Version Manager では開発の各段階におけるファイルを別々の物理的位置に置いたり、ファイルのプロモーションを行うために場所を移動する必要はありません。

Version Manager のアーカイブには、リビジョンとプロモーショングループとの間の論理的関連に関する情報が保存されています。Version Manager のデスクトップクライアントには、次の図に示すようにこの関連付けが表示されます。プロモーションモデルアイコンは、プロモーションモデル中でそれぞれのプロモーショングループに関連付けられたリビジョンを示します。

名前	チェックイン日時
bridge.clw	98/05/18 15:37:40
bridge.cpp	00/02/01 16:01:36
bridge.dsp	98/05/18 15:37:44
bridge.dsw	98/05/18 15:37:44

プロモーション モデルアイコン	プロモーショングループ	リビジョン
	Development	1.0

プロモーションモデルのルール

ルール 1 : プロモーションモデルには、1 つまたは複数の、最下位のグループを作成できる。

ルール 2 : プロモーションモデルが設定されている場合は、チェックアウトまたはロックするリビジョンのすべてに最下位のプロモーショングループが関連付けられている必要がある。プロモーションモデルに最下位のグループが 1 つしかない場合は、チェックアウトまたはロックされる各リビジョンに、そのプロモーショングループが自動的に関連付けられます。

プロモーションモデルに複数の最下位プロモーショングループが含まれており、[ファイル] - [プロパティ] - [ワークスペースの設定] タブでデフォルトの最下位プロモーショングループが設定されている場合は、特に別のグループを設定しないかぎり、その最下位プロモーショングループが使用されます。プロモーションモデルに複数の最下位プロモーショングループが含まれており、デフォルトのグループが設定されていない場合は、ダイアログボックスが表示され、プロモーショングループを選択するよう求められます。

たとえば、以下のようなプロモーションモデルが設定されているとします。

Development ⇒ QA_Test1 ⇒ Release

次の図に示すように、bridge.dsp のリビジョン 1.0 がチェックアウトされるとき、このリビジョンには最下位のグループである Development グループが自動的に関連付けられます。

名前	チェックイン日時
bridge.clw	98/05/18 15:37:40
bridge.cpp	00/02/01 16:01:36
bridge.dsp	98/05/18 15:37:44
bridge.dsw	98/05/18 15:37:44

プロモーショングループ	リビジョン
Development	1.0

ルール 3 : 最新のリビジョンには、最下位のグループが自動的に割り当てられる。 ファイルのそれ以降のリビジョンをチェックアウトすると、最下位のグループは最新のリビジョンに割り当てられません。

名前	チェックイン日時
bridge.clw	98/05/18 15:37:40
bridge.cpp	00/02/01 16:01:36
bridge.dsp	07/01/30 14:36:24
bridge.dsw	98/05/18 15:37:44

プロモーショングループ	リビジョン
Development	1.1

ルール 4 : リビジョンがその上位のプロモーショングループにプロモートされると、そのリビジョンと最下位のプロモーショングループとの間の関連付けは消滅する。 次の図に示すように、Development に関連付けられていたリビジョンがプロモートされると、このリビジョンには QA_Test1 グループが関連付けられ、Development グループとの関連付けは消滅します。

名前	チェックイン日時
bridge.clw	98/05/18 15:37:40
bridge.cpp	00/02/01 16:01:36
bridge.dsp	07/01/30 14:36:24
bridge.dsw	98/05/18 15:37:44

プロモーショングループ	リビジョン
QA_Test1	1.1

ルール 5 : リビジョンがより上位のプロモーショングループに関連付けられている場合は、チェックアウト時に最下位のプロモーショングループも関連付けられる。 前述のように、ロックによりチェックアウトされるリビジョンには、必ず最下位のプロモーショングループが関連付けられます。した

がって、QA_Test1 などの、より上位に関連付けられたリビジョンがチェックアウトされる場合は、最下位のプロモーショングループとの関連付けも追加されます。

名前	チェックイン日時
bridge.clw	98/05/18 15:37:40
bridge.cpp	00/02/01 16:01:36
bridge.dsp	07/01/30 14:36:24
bridge.dsw	98/05/18 15:37:44

プロモーショングループ	リビジョン
Development	1.1
QA_Test1	1.1

この段階で、このリビジョンは QA_Test1 と Development の両方のグループに関連付けられています。ただし、新規のリビジョンがチェックアウトされる時は、Development グループのみが関連付けられ、その前のリビジョンは QA_Test1 グループと関連付けられたままになります。

ルール 6 : 1 つのプロモーショングループが一度に関連付けられるのは、1 つのリビジョンのみ。 1 つのリビジョンは、ルール 5 で説明したように複数のプロモーショングループと関連付けられます。ただし、1 つのプロモーショングループは、同時にアーカイブ中の 1 つのリビジョンとのみ関連付けられます。したがって、有効なプロモーションモデルに最下位のプロモーショングループが 1 つしかない場合に、ほかのリビジョンが最下位のグループとすでに関連付けられているときにリビジョンをチェックアウトしようとしても、そのリビジョンはチェックアウトできません。たとえば、アーカイブ中にリビジョン、1.0 から 1.5 までが存在し、最下位のプロモーショングループがリビジョン、1.5 と関連付けられている場合は、リビジョン、1.4 やその他のリビジョンはチェックアウトできません。

この場合に行える作業は、次のいずれかです。

- 最下位のグループに関連付けられたリビジョンをプロモートする。
- 別の最下位のグループを定義する。

プロモーションモデルの定義

プロモーションモデルは、マスターコンフィグレーションファイルと、プロジェクトまたはローカルなコンフィグレーションファイルの両方で定義できます。デスクトップクライアントでは、この場合、プロジェクトデータベースに対してプロモーションモデルが定義され、プロジェクト / サブプロジェクトに対してはほかのプロモーションモデルが定義されたと考えられます。

両方のタイプのコンフィグレーションファイルにおいてプロモーションモデルが定義されている場合は、これらのモデルが共同で機能します。プロジェクトまたはローカルコンフィグレーションファイル中のプロモーションモデル定義は、マスターコンフィグレーションファイル中のプロモーションモデル定義より優先されることはありません。たとえば、マスターコンフィグレーションファイルで次のようなプロモーションモデルを定義するとします。

Dev ⇒ QA ⇒ Production

また次のようなプロモーションモデルをプロジェクトまたはローカルコンフィグレーションファイルで定義するとします。

Dev1⇒ QA

この場合、最下位のプロモーショングループには Dev と Dev1 の 2 つが存在し、それぞれ QA にプロモートされます。QA はマスターコンフィグレーションファイルに定義された Production にプロモートされます。QA をプロジェクトまたはローカルコンフィグレーションファイル中の Production プロモーションパスに複製することはできません。これを行うとこのプロモーションモデルは無効となり、プロジェクトを開く前にコンフィグレーションを修正（重複を削除）する必要があります。

マスターコンフィグレーションファイル中でプロモーションモデルを定義し、次にプロジェクトまたはローカルコンフィグレーションファイル中でもう 1 つのプロモーションモデルを定義した場合は、マスターコンフィグレーションファイルと、プロジェクトまたはローカルコンフィグレーションファイル中の定義が衝突しないよう注意が必要です。たとえば、マスターコンフィグレーションファイル中のプロモーションモデルが前述のとおりである場合、プロジェクトまたはローカルコンフィグレーションファイル中では次のように定義することはできません。

Dev ⇒ Production

これは、Version Manager に対してまず Dev を QA にプロモートし、次に Dev を Production にプロモートするよう指示することを意味するため、衝突が発生します。

デスクトップクライアントを使用する場合

プロモーションモデルを定義するには、「プロジェクトの設定」権限を割り当てられている必要があります。プロジェクトデータベースまたはプロジェクトにはコンフィグレーションファイルが関連付けられており、プロジェクト内でプロモーションモデルを定義する場合には、マスターコンフィグレーションファイルにおいて Promote オプション（ディレクティブ）が許可されている必要があります。

デスクトップクライアントにおいて、それぞれプロモーションモデルの定義を含むプロジェクトの階層が存在する場合は、デスクトップクライアントには作業中のプロジェクトに関して有効なプロモーションモデルが表示されます。前述の例を使用すれば、プロジェクトデータベース（マスターコンフィグレーションファイル）に対して作業を行う場合のデスクトップクライアント表示は次のようになります。



プロジェクトに対して作業を行う場合は、デスクトップクライアントにはプロジェクトデータベース中で定義されたプロモーションモデルと、プロジェクト中で定義されたプロモーションモデルが表示されます。



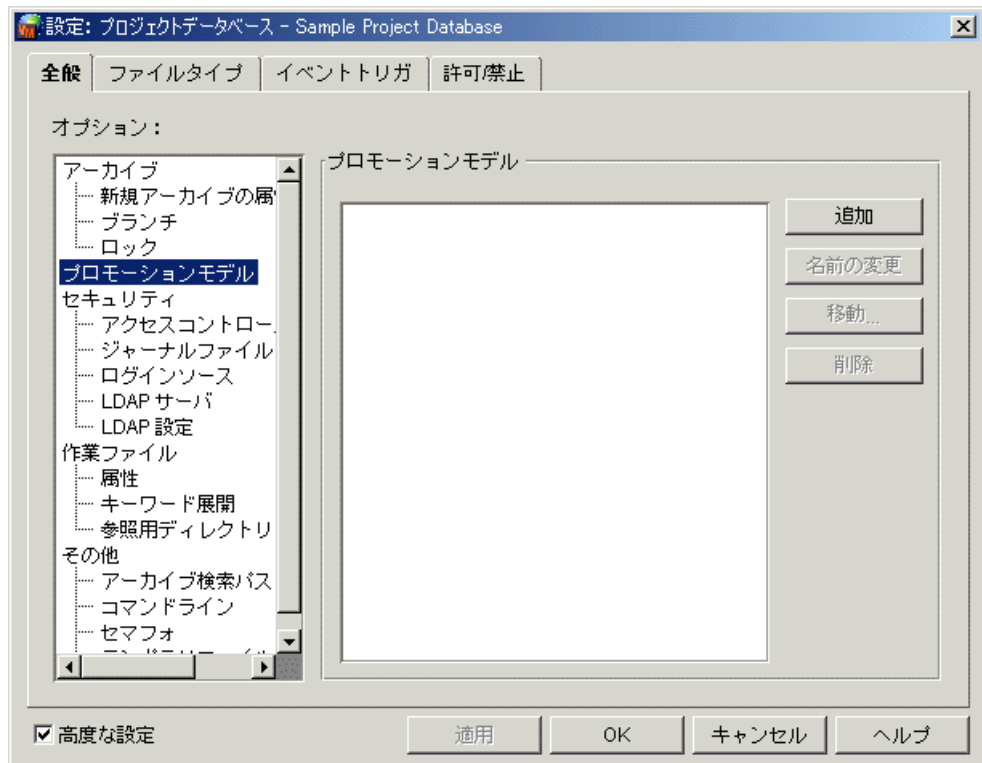
プロモーションモデルを定義するには

- 1 プロモーションモデルを定義するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 〈オプション〉リストから「プロモーションモデル」を選択します。右側に〈プロモーションモデル〉表示部が表示されます。プロジェクトに対してプロモーションモデルを定義する場合に、親プロジェクトまたはプロジェクトデータベースにプロモーションモデルが定義されていれば、表示されるプロモーションモデルには親プロジェクトのプロモーションモデル定義が含まれま

す。ただしプロモーションモデルのうち、親プロジェクト中で定義された部分の名称を変更したり、削除することはできません。これを行うには手順 1 で親プロジェクトを選択する必要があります。



注 デスクトップクライアントでは子プロジェクトに対して作業を行っているときに、親プロジェクトのプロモーショングループを選択して【削除】を使用できますが、この変更は保存されません。また、デスクトップクライアントでは親プロジェクト中で定義されたプロモーショングループに対して【名前の変更】も使用できます。この場合、名称を変更されたプロモーショングループは現在のプロジェクトのプロモーションモデルに追加され、親プロジェクトのプロモーションモデルが名称を変更されます。この動作は、衝突を引き起こす可能性があります。親プロジェクト中でプロモーションモデルが定義された子プロジェクトに対して作業を行う場合は、子プロジェクト中で定義されたプロモーションモデルに対してのみ変更を行い、階層上でそれよりも上のプロモーションモデルには変更を行うべきではありません。



- 4 次のいずれかの作業を行い、プロモーショングループを追加します。
 - プロモーショングループが存在しない場合は、【追加】をクリックして最上位のグループを定義します。
 - 選択されたグループの下に新しいグループが追加されます。既存のプロモーショングループが存在する場合は、グループを 1 つ選択して、【追加】をクリックします。
 - 最下位のプロモーショングループを追加するには、既存の最下位のグループを選択し、【追加】をクリックします。次にこのグループを移動して、既存の最下位のグループと同じレベルに置きます。プロモーショングループの移動については、手順 6 を参照してください。
- 5 いずれかのプロモーショングループの名称を変更するには、そのプロモーショングループを選択し、【名前の変更】をクリックして、新しい名前を入力します。
- 6 プロモーショングループをプロモーショングループ階層中で上下に移動するには、プロモーショングループを選択して【移動】をクリックします。移動しようとするプロモーショングループの親グループを選択するためのダイアログが表示されます。

- 7 プロモーショングループのいずれかを削除するには、そのプロモーショングループを選択し、【削除】をクリックします。
- 8 【OK】をクリックします。

コマンドラインインターフェイスを使用する場合

主に Version Manager のコマンドラインインターフェイスを使用する場合は、テキストエディタを使用してコンフィグレーションファイルを編集することもできます。また、デスクトップクライアントを使用して Version Manager を設定し、コンフィグレーションファイルを編集することもできます。デスクトップクライアントによって設定したコンフィグレーションファイルは、コマンドラインインターフェイスでも有効です。

ローカルコンフィグレーションファイルを設定する場合は、マスターコンフィグレーションファイルで Promote ディレクティブを使用して、プロモーションモデルを定義する必要があります。

コンフィグレーションファイルに Promote ディレクティブを追加します。次に例を示します。

```
Promote Development QA
Promote QA Release
```

この例では、Development、QA、および Release の順で階層が設定されます。

Promote ディレクティブの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

プロモーションとライフサイクルの管理

プロモーションモデルのおもな目的は、ソフトウェア開発における設計段階から最終リリースまでのライフサイクルを正式に管理することです。このセクションでは、ライフサイクル管理のためのプロモーションモデルの使用例を示します。

シナリオ

次のような単純なライフサイクルがあるとします。

Development ⇒ QA ⇒ Release

このシナリオでは、Development、QA、および Release というライフサイクルに対応したプロモーショングループにより、プロモーションモデルが定義されます。このプロモーションモデルがある場合は、開発者は、Development プロモーショングループに関連付けられた、デフォルトでチップリビジョンとなるファイルをチェックアウトします。チップリビジョンには最下位のプロモーショングループが伴うため、ファイルをチェックインして戻すと、新しいリビジョンには Development プロモーショングループが関連付けられます。

次に開発者は製品のリリースに必要なすべての変更を行い、リビジョンは Development から QA にプロモートされます。次に仮想のアーカイブについて、変更を加える前と後の状況を示します。

変更前	変更後
Rev 1.5 Development	Rev 1.5 QA
Rev 1.4	Rev 1.4

変更前	変更後
Rev 1.3	Rev 1.3
Rev 1.2 Release	Rev 1.2 Release
Rev 1.1	Rev 1.1
Rev 1.0	Rev 1.0

ここで品質管理部門はソフトウェアのテストを開始し、開発者は次のリリースに対する作業を開始して、チェックアウトとチェックインを伴う変更を行います。たとえば、開発者がこの仮想アーカイブに2つのリビジョンを追加したとすれば、その結果は次のようになります。

Rev 1.7 Development
 Rev 1.6
 Rev 1.5 QA
 Rev 1.4
 Rev 1.3
 Rev 1.2 Release
 Rev 1.1
 Rev 1.0

品質管理部門がリビジョンをチェックアウトするときは、QA プロモーショングループを対象とするため、開発による新たな変更には留意する必要はありません。

その後に品質管理部門がバージョンの品質が十分と判断すると、QA に関連付けられたリビジョンは Release にプロモートされます。この段階で、この仮想アーカイブは次のようになります。

Rev 1.7 Development
 Rev 1.6
 Rev 1.5 Release
 Rev 1.4
 Rev 1.3
 Rev 1.2
 Rev 1.1
 Rev 1.0

ここで、リリースマネージャは最終的なソフトウェア製品を作成します。このマネージャがリビジョンをチェックアウトするときには、Release プロモーショングループを対象とし、開発者が行う変更には影響を受けません。

プロモーションと並行開発

このセクションでは、並行開発をコントロールするための、プロモーションモデルの使用方法について説明します。このセクションの最後には、並行開発をコントロールするためのプロモーションモデルの定義シナリオを2つ示しています。

並行開発をコントロールするためのプロモーションモデルの使用

並行開発とは、開発における主ラインから、別のラインによる開発を分岐することを意味します。これは通常、次の場合に行われます。

- バグ修正など、開発の主ラインにすぐには影響を与えたくない変更を行う場合。たとえば開発作業を継続しながら、リリースに向けたバグの修正を行う場合が該当します。

- 製品の複数のバージョンに対して同時に作業を行う場合（たとえばマルチプラットフォームによる開発）。

プロモーションモデルが存在する場合に、同一ファイルの 2 つのバージョンについて並行に開発を行うには、それぞれのブランチに対して最下位のグループを作成する必要があります。

並行開発においてプロモーションモデルを使用することの利点は、1 つのリビジョンから作成可能なブランチの数を制限できることです。アーカイブではプロモーションに関連付けられるリビジョンは 1 つだけであり、リビジョンのチェックアウトは常に最下位のグループにおいて行わねばならないため、プロモーションモデル内に定義する最下位のグループの数によって、Version Manager がリビジョンに対して許可するロックの数が決定されます。したがって、定義した最下位のグループの数により、リビジョンから作成可能なブランチの数が決まります。たとえば、最下位のグループを 2 つ定義したとすれば、主ブランチ（トランク）と 1 つの並行ブランチの合計 2 つのブランチまでを作成できます。

シナリオ 1：緊急のバグ修正のためのプロモーショングループの定義

このシナリオは、251 ページの『プロモーションとライフサイクルの管理』で示されたシナリオを発展させたものです。これは、緊急の場合にバグ修正を行えるよう、プロモーションモデルを設定する方法の 1 つを示します。

顧客が新規リリース（前述の仮説アーカイブのリビジョン 1.5 を含むもの）においてバグを発見したとします。リリースマネージャはこのバグを修正し、できるだけ早くパッチを送付したいと考えています。

この状況に対応するため、リリースマネージャは Bug_fix というプロモーショングループを作成します。これは次に示すように、最上位のグループである Release に直接プロモートされます。



バグの修正のため、開発者は Release に関連付けられたリビジョンをチェックアウトします。開発者がリビジョンをチェックアウトする場合には、2 つの最下位グループの 1 つである Bug_fix プロモーショングループを選択する必要があります。開発者はバグを修正し、このリビジョンをブランチリビジョン 1.5.1.0 としてチェックインします。

リビジョン	プロモーショングループ	リビジョン
1.* (トランク)	Release	1.5
1.5	Bug_fix	1.5.1.0
1.4		
1.3		
1.2		
1.1		
1.0		
1.5.1.*		

開発者はただちにこのリビジョンを Release にプロモートし、リリースマネージャは Release プロモーショングループに基づいてアプリケーションを再構築します。

プロモーショングループ	リビジョン
Release	1.5.1.0

その後で、開発者はこのブランチを Development グループにおいてトランクにマージします。これにより、このバグ修正がそれ以降のリリースに含まれることが保証されます。

シナリオ 2：マルチプラットフォーム開発のためのプロモーショングループ定義

このセクションでは、製品に別のバージョンを作成するため、プロモーションモデルを使用してブランチに対応する場合について説明します。



注 ブランチをトランクにマージして戻す場合、あるいはブランチかトランクのいずれかに大きな変更を行う場合には、開発の各ラインに対して別々のアーカイブを作成してください。

次に示すプロモーションモデルは、マルチプラットフォームによる開発に対応しています。各アーカイブにおいて、トランクは Windows 版の製品の開発、ブランチは UNIX 版の開発に使用されます。

WinDev ⇒ WinTest ⇒ Production

UnixDev ⇒ UnixTest

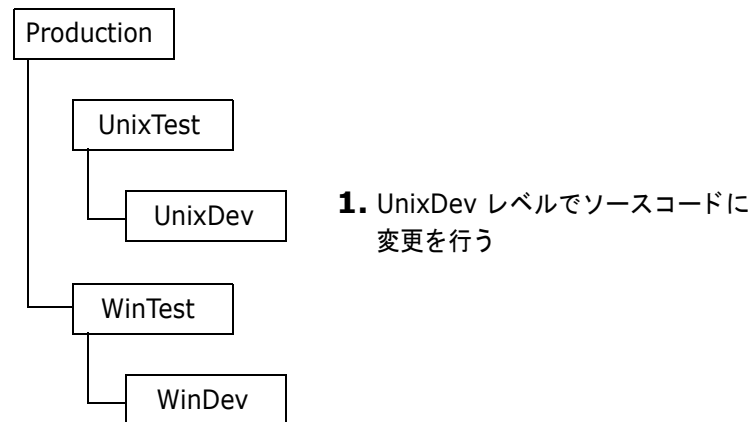
ここでは、UnixTest が Production にプロモートされることはありません。

この目的のためにプロモーションモデルを設計することは、UNIX 版に対する変更がごくわずかであり、定期的にマージが行われる場合にのみ効果的です。

以降のステップでは、次の作業を行うための手順を説明しています。

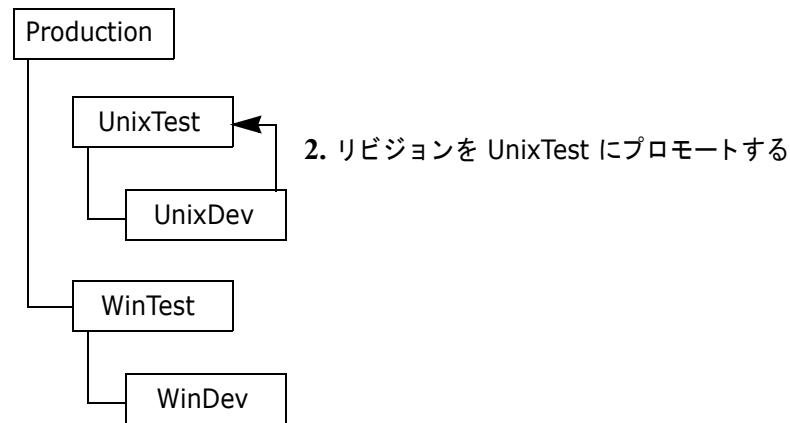
- ブランチに変更を加える
- テスト作業に向けてプロモートする
- 変更箇所を、開発の主ラインにマージする
- 製造に向けてプロモートする

手順 1：開発者は、UnixDev レベルにおいてソースコードに変更を行い、UNIX 開発のためのブランチを作成します。ブランチ作成については、259 ページの第 8 章『ファイルのブランチとマージ』を参照してください。

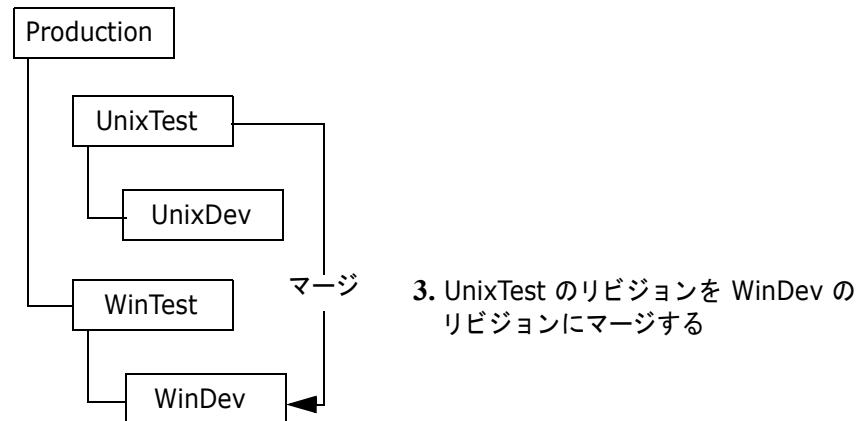


手順 2：開発者は単体試験のために、リビジョンを UnixTest にプロモートします。UnixTest に関連付けられたリビジョンは、最上位のグループにプロモートされることはありません。これにより、

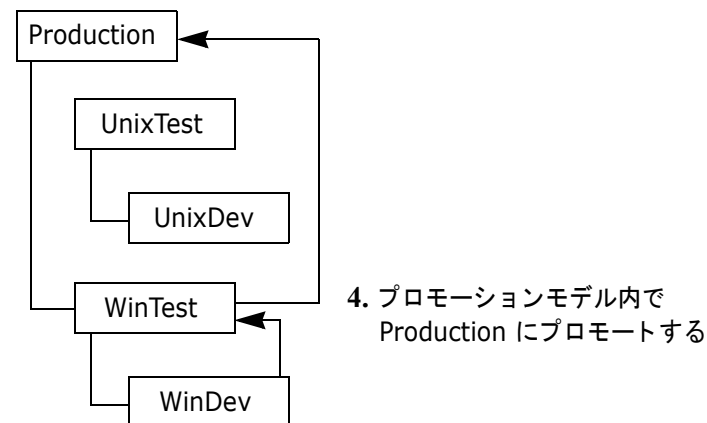
すべてのコードが Production に到達するまでにマージされ、適切なテストを経てプロモートされることが保証されます。



手順 3 : 開発者は UnixTest に関連付けられたリビジョンを、WinDev プロモーションレベルのリビジョンにマージします。マージについては、[259 ページの第 8 章『ファイルのブランチとマージ』](#)を参照してください。 .



手順 4 : リビジョンを WinDev から WinTest に、そして、最終的に Production にマージします。



ユーザによるプロモートの制限



注 Version Manager のセキュリティの仕組みを熟知していない場合は、このセクションに進む前に 201 ページの『セキュリティの使用』を参照してください。ユーザによるプロモートを制限するには、プロジェクトデータベースに対してプロモーションモデルが有効になっている必要があります。

プロモーショングループによる制限

このセクションでは、プロモーショングループをセキュリティ機能と統合して、リビジョンの 1 プロモーショングループからその次へのプロモートを制御する方法を説明します。プロモーショングループによる制限機能を使用するには、次の操作を行う必要があります。

- 1 カスタム権限セットを定義します。
- 2 このカスタム権限セットを特定のプロモーショングループのみに制限します。
- 3 このカスタム権限セットをユーザまたはアクセスリストに割り当てます。

こうすると、該当するユーザは、割り当てられたプロモーショングループからその次のグループへのプロモートのみが可能となります。このように、ユーザがプロモート可能なレベルを次のレベルのみに制限しておくと、プロモーションレベルの階層構造の中で、確実に 1 レベルずつプロモートされるようにすることができます。



注 「プロモーショングループの作成」、「プロモーショングループの変更」、および「プロモーショングループの削除」の各権限を持つ管理者は、プロモーションモデルやその設定による制限を受けません。

例 たとえば、Dev、QA、Prod の順に上がっていくプロモーションモデルを作成したとします。そして、カスタム権限セットを作成して「次のプロモーショングループへのプロモート」権限を有効にし、このカスタム権限セットを Dev プロモーショングループに制限します。こうすると、このカスタム権限セットを割り当てられたユーザがプロモートできるのは、Dev から QA へのみとなり、Dev から直接 Prod にプロモートされるのを防ぐことができます。

ユーザによるプロモートの制限に使用されるカスタム権限セットには、1 件の制限のみを定義してください。それ以外の権限を割り当てる場合は、デフォルトのカスタム権限セットを使用するか、または新しいカスタム権限セットを作成してください。



注 ユーザのほかのカスタム権限セットでは、「次のプロモーショングループへのプロモート」権限が無効になっており、かつ「プロモーショングループの作成」、「プロモーショングループの変更」、および「プロモーショングループの削除」などの管理機能を持っていないことが必要です。これらの条件を満たしていない場合は、プロモーショングループによるプロモートを制限するために定義されたカスタム権限セットが無効となります。

複数の最下位レベルプロモーショングループの使用

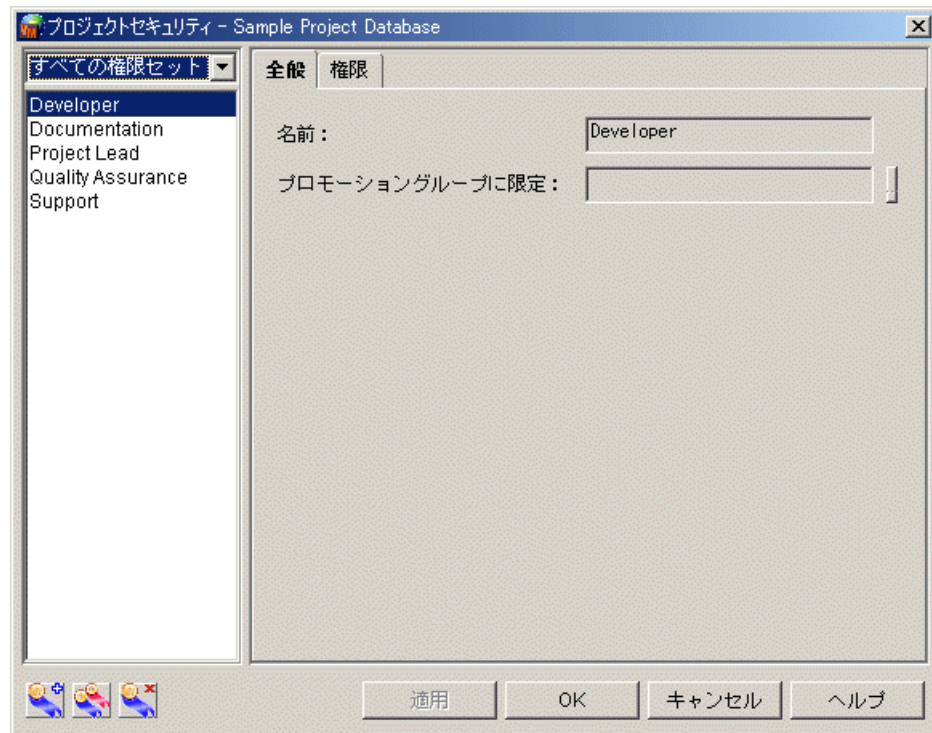
いったんバージョン管理ファイルにプロモーションモデルが適用されると、チェックアウトされた各リビジョンは、デフォルトでプロモーションモデルで最下位レベルのプロモーショングループに割り当てられます。複数の最下位プロモーショングループを使用している場合は、その中の一つを Lock のみに制限する必要があります。

デスクトップクライアントを使用する場合

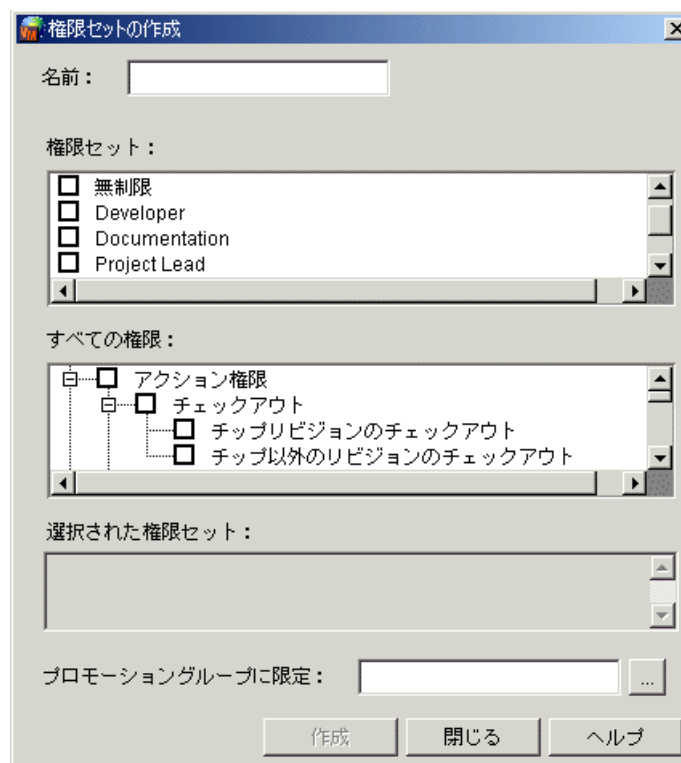
プロモーショングループによってユーザによるプロモートを制限するには

- 1 権限セットを定義しようとするアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。

- 2 [管理] - [セキュリティ] - [権限セット] を選択します。[プロジェクトセキュリティ] ダイアログボックスが開き、[全般] タブがアクティブになっています。



- 3 [新規作成] アイコンをクリックします。[権限セットの作成] ダイアログボックスが表示されます。



4 カスタム権限セットの名前を入力します。

この名前の先頭と末尾には、タブ記号およびスペースは使用できません。また、半角のかっこ (())、引用符 (')、二重引用符 (")、コロン (:)、円記号 (¥)、アスタリスク (*) も使用できません。

5 〈すべての権限〉フィールドで「次のグループへプロモート」を選択します。これ以外の権限が選択されていないことを確認してください。

権限セット名の前にあるボックスに「+」のついた権限は有効になっています。「-」は、禁止権限であるプロジェクト権限のみにつけられ、その権限が禁止されていることを示します。

権限の選択状況を変更するには、権限名の隣のチェックボックスをクリックします。親権限をクリックすることにより、関連する権限のセット全体で有効または無効にすることができます。たとえば、「アクション権限」の隣のチェックボックスをクリックして「+」記号を表示することにより、すべてのアクション権限を割り当てることができます。

6 〈プロモーショングループによる制限〉テキストボックスに、カスタム権限セットを制限するためのプロモーショングループを入力します。

7 【作成】をクリックして次に【閉じる】をクリックし、この権限セットを作成して〔プロジェクトセキュリティ〕ダイアログボックスに戻ります。

8 【OK】をクリックします。

9 ユーザまたはアクセスリストにこの権限セットを割り当てます。

次の項目については、それぞれの参照先を参照してください。

- プロモーションモデルの定義については、[248 ページの『プロモーションモデルの定義』](#)を参照してください。
- アクセスリストグループの定義については、[218 ページの『アクセスリストグループの定義』](#)を参照してください。
- アクセスリストの定義については、[226 ページの『アクセスリストの定義』](#)を参照してください。

コマンドラインインターフェイスを使用する場合

プロモーショングループごとにアクセスを制限するには、1 つまたは複数のプロモーショングループに限定されるよう、アクセスコントロールテキストファイル中のカスタム権限セットを定義する必要があります。[224 ページの『カスタム権限セットの定義』](#)を参照してください。

第 8 章

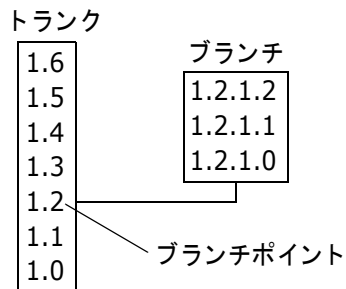
ファイルのブランチとマージ

ブランチ	260
自動ブランチ	261
ブランチのための多重ロックの使用	266
マージ	268
自動マージ	269

ブランチ

ブランチとは、トランク（主ライン）上のリビジョンまたはほかの開発ブランチから枝分かれした、1 つまたは複数のリビジョンによって構成される、別ラインによる開発を意味します。図 8-1 には、ブランチを使用したファイルの別バージョン開発を、トランクまたはほかのブランチ上で作業を行うほかの開発者と平行して行っているところを示します。

図 8-1 : ブランチ



ブランチを作成する主な理由は次のとおりです。

- ほかのプラットフォームのためのバージョンを開発する場合。たとえば、Windows アプリケーションのためのソースコードを保存するアーカイブがいくつかある場合に、このアプリケーションの UNIX バージョンを作成するには、各アーカイブにおいてブランチを行います。ブランチとプロモーショングループの例については、254 ページの『シナリオ 2 : マルチプラットフォーム開発のためのプロモーショングループ定義』を参照してください。
- トランクにおける開発を中断することなくバグを修正する場合。バグを発見したがバグ個所以外のソースコードファイルについては開発を続ける場合、バグを含むリビジョンからブランチを作成し、バグを修正し、それをテストするという一連の作業をトランク上の開発を妨げることなく行えます。その後、この修正箇所をトランク上の最新バージョンとマージできます。
- 基本となる製品を作成し、次に大口顧客に向けてその製品をカスタマイズする場合。

ブランチを利用すると、多くの開発者が同じファイルの別のリビジョンで並行開発を続けることができます。また、1 人の開発者がトランク開発とさまざまなブランチの両方で作業することもできます。

ブランチによる分岐は、トランクから、またはほかのブランチから行えます。トランクからの最初のブランチリビジョンには、トランクリビジョンの 2 桁のリビジョン番号に、2 桁のブランチリビジョン番号を続けたものが付けられます。ブランチが開始されるリビジョンは、ブランチポイントと呼ばれます。

たとえば、260 ページの図 8-1 : ブランチでは、リビジョン 1.2 がブランチポイントになっているため、最初のブランチリビジョンの番号は 1.2.1.0 になります。トランクリビジョンと同様、Version Manager では新しいブランチリビジョンごとに 0.1 が追加されます。したがって、このブランチの以降の番号は 1.2.1.1、1.2.1.2 などになります。

Version Manager では、ブランチ全体は最初の 3 つの番号によって識別されます。たとえば、ブランチ 1.2.1 には、リビジョン 1.2.1.0、1.2.1.1、1.2.1.2 などが含まれ、ブランチ 1.3.1 には、リビジョン 1.3.1.0、1.3.1.1 などが含まれます。

ブランチが作成される場合

Version Manager では、次の場合にブランチが作成されます。

- ロックされた、チップ以外のリビジョンをチェックインした場合。リビジョンのチェックインについては『Serena ChangeMan Version Manager ユーザガイド』を参照してください。

- ロックされたチップリビジョンをチェックインするときに、強制的にブランチさせる場合。コマンドラインインターフェイスでは、PUT -FB コマンドを使用してブランチを強制します。デスクトップクライアントでは、[チェックイン] ダイアログボックスの〈強制的にブランチさせる〉チェックボックスを選択します。詳細については『Serena ChangeMan Version Manager ユーザガイド』を参照してください。
- 自動ブランチを行うよう Version Manager を設定した場合。次のセクションを参照してください。
- 二次ロックによりリビジョンをチェックインした場合。これは、複数のユーザが同一のリビジョンをロックしてチェックアウトしたことを意味します。最初にリビジョンをロックするユーザには、そのリビジョンに対する一次ロックが与えられます。ほかのユーザがそのリビジョンをロックする場合は、それぞれのリビジョンをブランチとしてチェックインする必要があります。リビジョンを複数回ロックできるのは、プロジェクトまたはプロジェクトデータベースが多重ロックを使用するように設定されている場合のみです。266 ページの『ブランチのための多重ロックの使用』を参照してください。

自動ブランチ

自動ブランチが設定されていると、トランクリビジョンから自動的にブランチが作成されます。その場合、デフォルトでは、作業はすべてそのブランチのチップリビジョンに対して行います。

自動ブランチが設定されていない場合は、チップ以外のリビジョンのロック、またはチップリビジョンをチェックインするときの強制ブランチのいずれかの方法により、手動でブランチを作成する必要があります。また、ブランチチップリビジョンをチェックアウトするときには、それを指定し、新しいリビジョンをチェックインするときにバージョンラベルを再度割り当てる必要があります。

自動ブランチの設定

自動ブランチオプションを設定する前に、ブランチを行うリビジョンに対して 2 つのバージョンラベルを割り当てる必要があります。ブランチポイントを示すため、該当するリビジョンに固定したバージョンラベルを割り当てます。次にこのリビジョンに対してもう 1 つの固定バージョンラベルを割り当てます（これはブランチが作成されたときに浮動ラベルとなります）。

バージョンラベル	リビジョン
Rel1.5 Branch	1.1
Rel1.0 Base	1.1

自動ブランチを設定するには、ブランチ設定のオプションを 3 つとも指定する必要があります。これらのオプションはブランチの起点と終点を定義し、作業の対象とするデフォルトのリビジョンを指定します。

- 「ベースバージョン」は、ブランチを開始するリビジョンを示すために割り当てるバージョンラベルを指定します。これは前述の例では Rel 1.0 Base になります。
- 「ブランチバージョン」は、ブランチのチップに割り当てるバージョンラベルを指定します。これは前述の例では Rel 1.0 Branch になります。当初はこれはブランチを行うリビジョンになります。以降のブランチリビジョンをチェックインすると、このラベルは自動的にブランチのチップリビジョンに割り当て直されます。
- 「デフォルトバージョン」は、ブランチバージョンオプション（前述の例では Rel 1.0 Branch）に対して指定したバージョンラベルを示します。このバージョンラベルは、どのバージョンに対して作業を行うかを Version Manager に伝えます。

ベースバージョンとブランチバージョンに対して指定されたバージョンラベルが同一のリビジョンを示している場合は、ファイルをチェックインすると自動的にそのリビジョンからブランチが開始されます。デフォルトバージョンにはブランチバージョンと同じバージョンラベルを設定するため、チップリビジョンのチェックアウトは、自動的にそのブランチ上で行われます。

デスクトップクライアントを使用する場合

自動ブランチを設定するには、ユーザに「プロジェクトの設定」権限が割り当てられており、プロジェクトデータベースまたはプロジェクトにコンフィグレーションファイルが関連付けられている必要があります。またプロジェクト内に自動ブランチを設定する場合は、マスターコンフィグレーションファイルでベースバージョン、ブランチバージョン、およびデフォルトバージョンが許可されている必要があります。

プロジェクトデータベースまたはプロジェクトに関連付けられたコンフィグレーションファイル中にブランチオプションを設定する場合は、同一の自動ブランチ定義がそのデータベースまたはプロジェクト中のすべてのプロジェクト / サブプロジェクトに設定されます。あるいは、ワークスペース中に別の自動ブランチ定義を設定することも可能です。このワークスペース中での設定は、マスターコンフィグレーションファイルで「ブランチ」オプションを禁止していないかぎり、コンフィグレーションファイル中の設定より優先されます。[174 ページの『ワークスペースの使用』](#)を参照してください。

たとえば、片方のグループは Rel 1.0、もう片方は Rel 1.5 の同一のコードに対して作業を行っている、2つの開発グループがあるとします。プロジェクトの初期には、Rel 1.0 と Rel 1.5 にそれぞれ加えられる変更は、同一アーカイブ中に共存します。Rel 1.5 には関係のない機能を Rel 1.0 に追加する必要が生じた段階で、ブランチを開始します。この製品のプロジェクトデータベースの名前は SoftwareGenius です。Rel 1.0 のプロジェクトリーダーはすべてのアーカイブを検討し、Rel 1.0 Base というバージョンラベルをブランチポイント（ブランチが開始されるリビジョン）に割り当てます。次にリーダーは同一のリビジョンに Rel 1.0 Branch Tip というバージョンラベルを割り当てます。

Rel 1.0 のためのバージョンラベルが割り当てられたら、プロジェクトデータベースのコンフィグレーションファイルに自動ブランチオプションを追加します。プロジェクトリーダーはベースバージョンとして Rel 1.0 Base を、ブランチバージョンおよびデフォルトバージョンとして Rel 1.0 Branch を割り当てます。

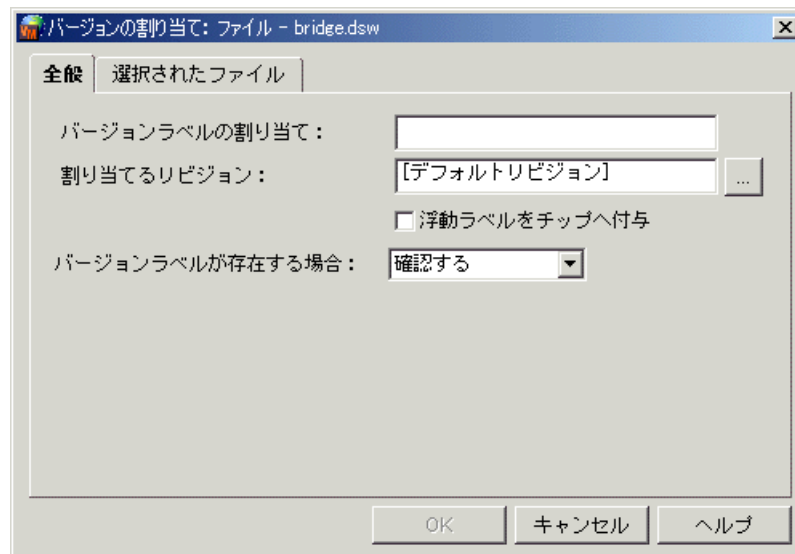
SoftwareGenius プロジェクトデータベースには、ワークスペースを 2 つ作成する必要があります。トランクのチップリビジョンをデフォルトリビジョンとする Rel 1.0 ワークスペースと、ブランチのチップリビジョンをデフォルトリビジョンとする Rel 1.5 ワークスペースです。これらのワークスペースが作成されている状態では、Rel 1.0 の開発者がコードの修正を行うときに自分のワークスペースを Rel 1.0 ワークスペースに設定します。チェックアウトするリビジョンを選択すると、バージョンラベル、Rel 1.0 Branch を持つリビジョンがデフォルトでチェックアウトされます（この開発者はブランチに対して作業を行います）。

同じ開発者が Rel 1.5 のコードに簡単な変更を行う場合は、この開発者は自分のワークスペースを Rel 1.5 に設定します。チェックアウトするリビジョンを選択すると、バージョンラベル、Rel 1.0 Base を持つリビジョンがデフォルトでチェックアウトされます（この開発者はトランクに対して作業を行います）。

自動ブランチを設定するには

- 1 自動ブランチを設定するプロジェクトデータベースまたはプロジェクトを選択します。
- 2 ブランチを開始するバージョン管理ファイルごとに、リビジョンに固定バージョンラベルを割り当てます。
 - a 選択されたプロジェクトから、バージョンラベルを割り当てるバージョン管理ファイルを選択します。ファイルの別のリビジョンにバージョンラベルを割り当てる場合は、バージョン管理ファイルをそれぞれ個別に選択する必要があります。

- b [アクション] - [バージョンラベル] - [割り当て] を選択します。[バージョンラベルの割り当て] ダイアログボックスが表示されます。



- c <バージョンラベル> テキストボックスに、Rel1.0 Base などのバージョンラベル名を入力します。名前には、半角のアスタリスク (*)、コロン (:)、二重引用符 (")、プラス記号 (+)、およびマイナス記号 (-) を除き、どの文字でも使用できます。
- d <割り当てるリビジョン> テキストボックスにリビジョン番号を入力するか、[...] をクリックしてリビジョン番号を選択します。
- e 【OK】 をクリックします。

このバージョンラベルは後に、ベースバージョンとして定義されます。

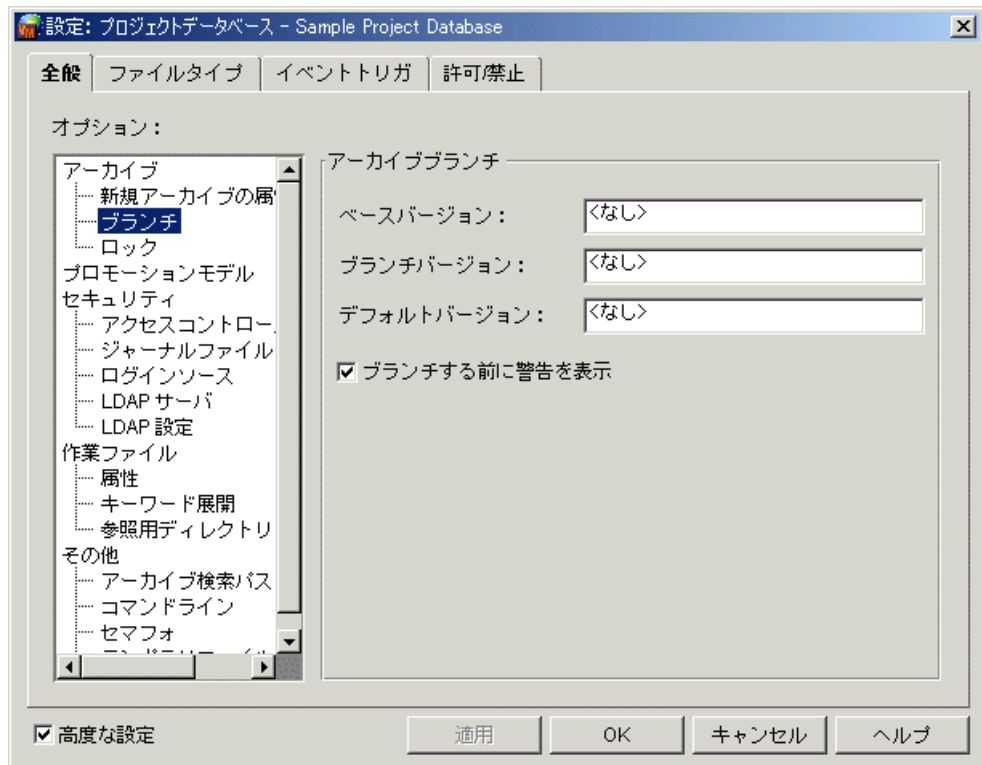
- 3 手順 2 で選択したバージョン管理ファイルのリビジョンに、もう 1 つの固定バージョンラベルを割り当てます。

- a バージョン管理ファイルを選択します。
- b [アクション] - [バージョンラベル] - [割り当て] を選択します。[バージョンラベルの割り当て] ダイアログボックスが表示されます。
- c <バージョンラベル> テキストボックスに、Rel1.5 Branch などのバージョンラベル名を入力します。名前には、アスタリスク (*)、コロン (:)、二重引用符 (")、プラス記号 (+)、およびマイナス記号 (-) を除き、どの文字でも使用できます。
- d <割り当てるリビジョン> テキストボックスにリビジョン番号を入力するか、[...] をクリックしてリビジョン番号を選択します。
- e 【OK】 をクリックします。

このバージョンラベルは後に、ベースバージョンとして定義されます。

- 4 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 5 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。

- 6 <オプション> リストから、「アーカイブ」－「ブランチ」を選択します。<アーカイブのブランチ> グループボックスが右側に表示されます。



- 7 <ベースバージョン> テキストボックスに、手順 2 で割り当てたバージョンラベル（例：Rel1.0 Base）をベースバージョンとして入力します。
- 8 <ブランチバージョン> テキストボックスに、手順 3 で割り当てたバージョンラベル（例：Rel1.5 Branch）をブランチバージョンとして入力します。
- 9 <デフォルトバージョン> テキストボックスに、手順 3 で割り当てたバージョンラベル（例：Rel1.5 Branch）をデフォルトバージョンとして入力します。
- 10 【OK】 をクリックします。

Version Manager に自動ブランチを設定すれば、[アクション]－[マージの表示] を使用して、ブランチのチップリビジョンをトランクのチップリビジョンに自動的にマージできます。ブランチオプションとして設定したバージョンラベルから、Version Manager はブランチが行われたリビジョン、トランクのチップリビジョン、およびブランチのチップリビジョンを判断します。したがって、マージを行う場合にこれらの値を入力する必要はありません。



注 トランクにおける作業を再開するには、デフォルトバージョンとして、手順 2 で指定したバージョンラベル、たとえば Rel 1.0 Base を再度設定します。

コマンドラインインターフェイスを使用する場合

主に Version Manager のコマンドラインインターフェイスを使用する場合は、テキストエディタを使用してコンフィグレーションファイルを編集することもできます。また、デスクトップクライアントを使用して Version Manager を設定し、コンフィグレーションファイルを編集することもできます。デスクトップクライアントによって設定したコンフィグレーションファイルは、コマンドラインインターフェイスでも有効です。

ブランチのディレクティブ、BaseVersion、BranchVersion、および DefaultVersion は、マスターコンフィグレーションファイルとローカルコンフィグレーションファイルのどちらにおいても定義できます。ただし、ローカルコンフィグレーションファイルでディレクティブを定義するには、マスターコンフィグレーションファイルにおいてそのディレクティブが許可されていることが必要です。

自動ブランチを設定するには

- 1 ブランチを開始するアーカイブごとに、VCS -V コマンドを使用してリビジョンに固定バージョンラベルを割り当てます。次にその例を示します。

```
vcs -vRel 1.0_Base:1.1 CVS.JAVA
```

このバージョンラベルは、後にベースバージョンとして定義されます。

- 2 VCS -V コマンドを使用して、上記で選択されているリビジョンと同じものに固定バージョンラベルを割り当てます。

```
vcs -vRel 1.0_Branch:1.1 CVS.JAVA
```

このバージョンラベルは、後にブランチバージョンとして定義されます。これにより、これらのバージョンラベルは、自動的にブランチのチップリビジョンに再度割り当てられます（「浮動する」ともいいます）。

- 3 テキストエディタを使用して、マスターコンフィグレーションファイルまたはローカルコンフィグレーションファイル（vcs.cfg）を開きます。

- 4 BaseVersion ディレクティブを、手順 1 で割り当てたバージョンラベルに指定します。次に例を示します。

```
BaseVersion=Rel 1.0_Base
```

- 5 BranchVersion ディレクティブを、手順 2 で割り当てたバージョンラベルに指定します。次に例を示します。

```
BranchVersion=Rel 1.0_Branch
```

- 6 DefaultVersion ディレクティブを、手順 2 で割り当てたバージョンラベルに指定します。次に例を示します。

```
DefaultVersion=Rel 1.0_Branch
```

- 7 コンフィグレーションファイルを保存し、テキストエディタを終了します。

Version Manager に自動ブランチを設定すると、VMRG -A コマンドを使用して、ブランチチップをトランクチップに自動的にマージできます。Version Manager は、ブランチオプションとして設定したバージョンラベルから、どのリビジョンをトランクにマージすべきかを判断します。

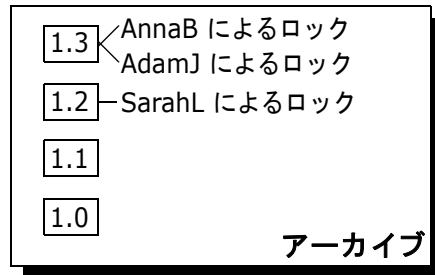


注 トランクにおける作業を再開するには、DefaultVersion ディレクティブを手順 1 で指定したバージョンラベル、たとえば Rel 1.0 Base に再度設定します。

ブランチのための多重ロックの使用

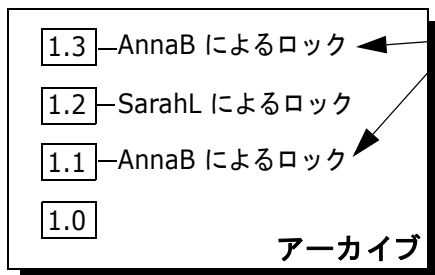
多重ロックは、グループが並行開発のためのパス（ブランチ）を維持することを想定して設計されています。オプションの設定により、次のような許可を設定できます。

- 単一リビジョンに対して多重ロックを設定（リビジョンごとに多重ロック）



アーカイブ中では各ユーザに許可されるロックは 1 つまでですが、ユーザは 1 つのリビジョンに対して複数のロックを設定できます。

- 単一のユーザが、アーカイブ中の複数のリビジョンに対してロックを設定（ユーザごとに多重ロック）



ユーザはアーカイブの複数のリビジョンにロックを設定できますが、どのリビジョンも 1 つのロックしか設定できません。

多重ロックの使用により、ブランチの作成が簡単になります。リビジョンごとに多重ロックが許可されている場合は、複数のユーザが同一のリビジョンにロックを設定できます。このオプションは、ほかのユーザが作業を行っているリビジョンからブランチを作成する場合に使用します。たとえば、あるユーザがトランクリビジョンに対して作業を行い、別のユーザがそのリビジョンからのブランチに対して作業を行う場合などに使用できます。

ユーザごとに多重ロックが許可されている場合は、同一のユーザがアーカイブ中の複数のリビジョンに対してロックを設定できます。このオプションは、開発中のファイルの 2 つまたはそれ以上のラインに対して、1 人のユーザが作業を行う場合に使用します。たとえば、同一のユーザがファイルのトランクとブランチリビジョンに対して作業を行う場合などに使用できます。

複数のユーザが同じ 2 つの開発ラインにおいて作業を行う場合は、両方のオプションを同時に使用できます。

ファイルをチェックインしたときにリビジョンが 2 回以上ロックされている場合は、ロックを最初に作成したユーザ以外のユーザに対しては新しいブランチが作成されます。ブランチが作成される前に、予約したロックを指定するよう求められ、そのリビジョンがブランチとしてチェックインされません。

多重ロックの設定

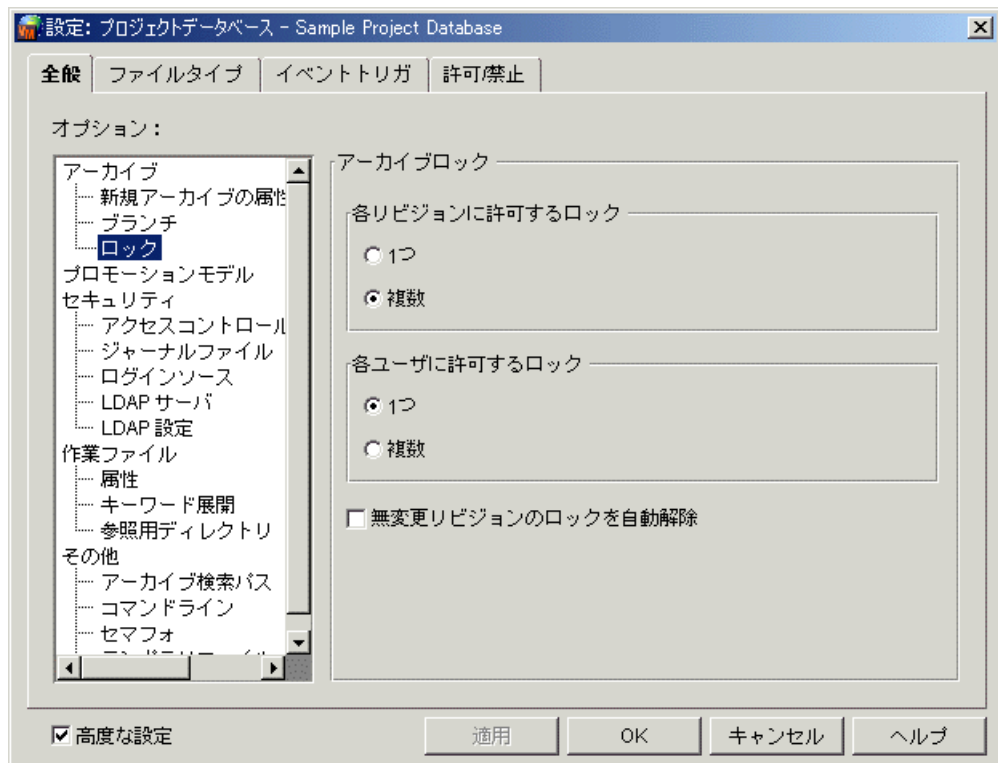
このセクションでは、デスクトップクライアントとコマンドラインインターフェイスを使用して多重ロックを設定するための手順について説明します。

デスクトップクライアントを使用する場合

多重ロックを設定するには、ユーザに「プロジェクトの設定」権限が割り当てられており、プロジェクトデータベースまたはプロジェクトにコンフィグレーションファイルが関連付けられている必要があります。また、プロジェクト内に多重ロックを設定する場合は、マスターコンフィグレーションファイルに「各リビジョンに許可するロック」および「各ユーザに許可するロック」のディレクティブが許可されている必要があります。

新しく作成されたアーカイブに多重ロックを設定するには

- 1 多重ロックを設定するプロジェクトデータベースまたはプロジェクトを選択します。この手順を完了すれば、プロジェクトデータベースまたはプロジェクト中のすべてのアーカイブに多重ロックを設定できるようになります。
- 2 [管理] - [詳細設定] を選択します。[プロジェクトの詳細設定] ダイアログボックスが表示され、[全般] タブがアクティブになっています。
- 3 <高度な設定> チェックボックスがまだ選択されていない場合は選択します。
- 4 <オプション> リストから、「アーカイブ」の下の「ロック」を選択します。<アーカイブのロック> グループボックスが右側に表示されます。



- 5 次のいずれかの作業を行います。
 - <各リビジョンに許可するロック> グループボックスで <複数> を選択し、リビジョンごとに多重ロックを許可します。
 - <各ユーザに許可するロック> グループボックスで <複数> を選択し、個々のユーザに対してアーカイブ中の複数のリビジョンを同時にロックすることを許可します。

6 【OK】をクリックします。



注 プロジェクトのすべてのアーカイブに対して多重ロックを設定した後、[管理] - [アーカイブ属性] を選択して [全般] タブの〈排他ロック〉を「はい」に設定し、プロジェクト中の特定のアーカイブについて複数ロックを禁止することができます。

既存のアーカイブに対して多重ロックを設定するには、[管理] - [アーカイブ属性] を選択する必要があります。185 ページの『[既存アーカイブの属性の変更](#)』を参照してください。

多重ロックに関してユーザが設定を変更できないようにするには、マスターコンフィグレーションファイル中の「ロック」オプション（ディレクティブ）を禁止します。116 ページの『[オプションの変更許可と変更禁止](#)』を参照してください。

コマンドラインインターフェイスを使用する場合

主に Version Manager のコマンドラインインターフェイスを使用する場合は、テキストエディタを使用してコンフィグレーションファイルを編集することもできます。また、デスクトップクライアントを使用して Version Manager を設定し、コンフィグレーションファイルを編集することもできます。デスクトップクライアントによって設定したコンフィグレーションファイルは、コマンドラインインターフェイスでも有効です。

多重ロックを設定するには

- 1 テキストエディタを使用して、マスターコンフィグレーションファイルまたはローカルコンフィグレーションファイル (vcs.cfg) を開きます。
- 2 以下のうち必要な作業を行います。
 - コンフィグレーションファイルに MultiLock ディレクティブをパラメータなしで記述し、リビジョンごとに多重ロックを許可し、かつ個々のユーザに対してアーカイブ中の複数リビジョンの同時ロックを許可します。
MultiLock
 - コンフィグレーションファイルに MultiLock ディレクティブを revision パラメータを指定して記述し、リビジョンごとに多重ロックを許可します。
MultiLock revision
 - コンフィグレーションファイルに MultiLock ディレクティブを user パラメータを指定して記述し、個々のユーザに対してアーカイブ中の複数リビジョンの同時ロックを許可します。
MultiLock user
- 3 コンフィグレーションファイルを保存し、テキストエディタを終了します。



注 アーカイブに対して多重ロックを設定した後、VCS + PE コマンドを使用して特定のアーカイブに対する複数ロックを禁止に設定できます。

マージ

マージとは、2 つ（またはそれ以上）の作業ファイルまたはリビジョンの相違点を比較し、それらの間の相違点を採用または排除し、かつ変更点を新しいファイルにまとめる作業を意味します。

マージはどのような 2 つのファイルまたはリビジョンの間においても行えますが、もっとも一般的には次のような場合に行われます。

- チップリビジョンの作業ファイルへのマージ。誤ってリビジョンをロックすることなく作業ファイルを更新し、次にほかのユーザがそれ以降にリビジョンをロックしたことが判明した場合は、マージを行うことにより、自分の編集箇所とほかのユーザの変更箇所をチェックイン時に安全に組み合わせられます。チップリビジョンの作業ファイルへのマージ手順については、『Serena ChangeMan Version Manager ユーザガイド』を参照してください。
- ブランチのトランクへのマージ。複数の開発パスを保守するため、または単に多重ロックを使用するためブランチを使用する場合に、ブランチチップをトランクチップに簡単にマージできます。自分のブランチを保守するために自動ブランチを設定した場合は、これらのリビジョンは自動的にマージできます。次のセクション [自動マージ](#) を参照してください。

Windows でデスクトップクライアントを使用する場合：Windows の Version Manager デスクトップクライアントでは、すべてのマージ作業に Merge Tool という別のユーティリティが使用されます。Windows では 3 方向マージ（1 つのベースファイルと 2 つのブランチファイル）を行えます。これは Serena ChangeMan Dimensions で使用するのと同じマージツールです。

UNIX およびコマンドラインインターフェイスでデスクトップクライアントを使用する場合：UNIX およびコマンドラインインターフェイスの Version Manager デスクトップクライアントでは、標準的な Version Manager マージユーティリティが使用されます。ここでは 2 方向マージ（1 つのベースファイルと 1 つのブランチファイル）を行えます。

自動マージ



注 自動マージを行うには、アーカイブに自動ブランチを設定しておく必要があります。

自動マージとは、以前に自動ブランチを設定したアーカイブにおいて、トランクにブランチをマージして戻す作業を意味します。自動ブランチが設定されているため、マージコマンドはブランチが行われた元のトランクチップリビジョンと、ブランチチップリビジョンを判別できます。このため、マージコマンドを実行するときにこれらの値を指定する必要はありません。

ファイルを自動的にマージするには、コマンドラインインターフェイスで `VMRG -A` コマンドを使用するか、デスクトップクライアントの [アクション] - [マージの表示] を選択します。リビジョンのマージの詳細については、『Serena ChangeMan Version Manager ユーザガイド』と『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

第 9 章

イベントトリガの使用

はじめに	272
Version Manager の処理イベント	272
カスタムツールの実行	274
イベントトリガのセットアップ	274
イベントトリガへのイベント情報の転送	276
イベントトリガの例	286

はじめに

イベントトリガは管理者によって指定されるプログラムで、特定の Serena ChangeMan Version Manager イベントが発生したときに実行されます。イベントとは次のものを指します。

- Version Manager における処理中に発生する、ファイルのチェックイン、バージョンラベルの割り当て、およびジャーナルファイルへの項目の追加などの特定のアクション。272 ページの表 9-1：処理イベントに Version Manager の処理イベントがすべてリストアップされています。
- ユーザが定義したツールバーアイコンのクリック。
- ユーザが定義したメニュー項目の選択。

Version Manager では、次のファイルもイベントトリガとしてサポートされます。

- スタンドアロンの Windows (.EXE) および UNIX 実行プログラム
- Windows バッチスクリプト (.BAT)
- Windows プログラム情報ファイル (.PIF)
- UNIX シェルスクリプト

Version Manager は、コマンドラインアプリケーションとグラフィカルアプリケーションの両方をサポートしています。グラフィカルアプリケーション以外のはすべて、シェルまたはコマンドインタープリタ、すなわち、Windows では COMSPEC、UNIX では SHELL を経由して実行されます。

イベントトリガは、新規ラベルの構文および作成者検証、プロジェクトがプロモートされるときの管理者へのメールメッセージ送付、リビジョンをチェックインした後のビルドスクリプトの実行、およびセキュリティ強化のための、保護されたファイルへのジャーナルファイル項目のコピーに使用されます。イベントトリガとセットアップの例は、286 ページの『イベントトリガの例』を参照してください。

Version Manager の処理イベント

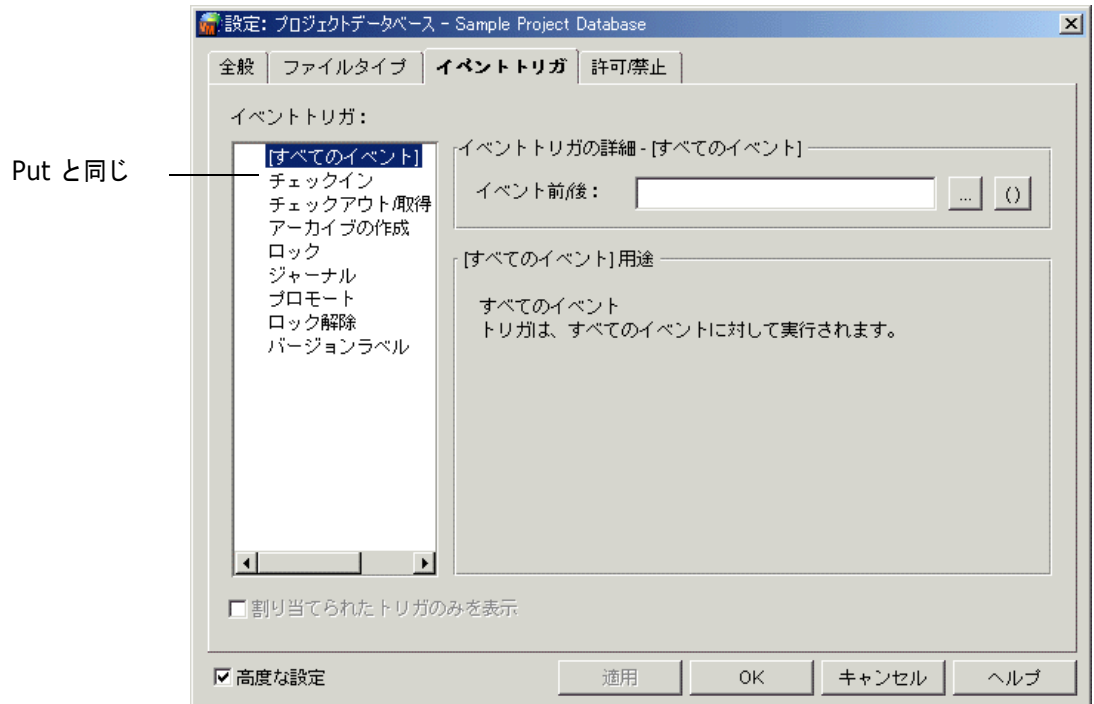
表 9-1 に Version Manager の処理イベントを定義します。この表に示されたイベントの名前はコマンドラインインターフェイス中でイベントトリガを定義するときにも使用され、またプロジェクトコンフィギュレーションファイルに保存するときにも使用されます。274 ページの図 9-1：デスクトップクライアントのイベント名に、デスクトップクライアントでイベントトリガの定義に使用されるダイアログボックスを示します。

表 9-1：処理イベント

イベント名	定義
AllEvents	この表にあるすべての Version Manager イベント。
PrePut	作業ファイルをチェックインする直前。このイベントは、Version Manager によるファイルのチェックインを妨げるものが何もない場合に発生します。
PostPut	作業ファイルのチェックインに成功した後。
UnconditionalPrePut	作業ファイルをチェックインする前。作業ファイルが Version Manager に読み込まれる前です。
PreGet	作業ファイルをチェックアウトする直前。

イベント名	定義
PostGet	作業ファイルのチェックアウトが正常に完了した後。
PrePromote	リビジョンをプロモートする直前。
PostPromote	リビジョンのプロモートが正常に完了した後。
PreVersionLabel	バージョンラベルを割り当てる直前。
PostVersionLabel	バージョンラベルを割り当てた後。このイベントは、ラベルを削除するときには発生しません。
PostJournal	ジャーナルファイルに項目が追加される時。
PreLock	リビジョンをロックする直前。このイベントは、リビジョンが同時にチェックアウトされるかどうかにかかわらず発生します。
PostLock	リビジョンのロック後。このイベントは、ロックと同時にリビジョンがチェックアウトされるかどうかにかかわらず発生します。
PreUnlock	<p>リビジョンのロックを解除する直前。次の場合には、変更されていない作業ファイルをチェックインするときにもこのイベントが発生します。</p> <ul style="list-style-type: none"> ■ ForceUnlock ディレクティブがコマンドラインインターフェイス中で有効になっている場合。 ■ デスクトップクライアントで、〈無変更のリビジョンのロックを自動解除〉オプションが設定されている場合。
PostUnlock	<p>リビジョンのロック解除後。次の場合には、変更されていない作業ファイルをチェックインするときにもこのイベントが発生します。</p> <ul style="list-style-type: none"> ■ ForceUnlock ディレクティブがコマンドラインインターフェイス中で有効になっている場合。 ■ デスクトップクライアントで、〈無変更のリビジョンのロックを自動解除〉オプションが設定されている場合。
PreCreateArchive	<p>新規のアーカイブを作成する直前。次の場合には、最初のリビジョンをチェックインする前にもこのイベントが発生します。</p> <ul style="list-style-type: none"> ■ AutoCreate ディレクティブがコマンドラインインターフェイス中で有効になっているとき。 ■ デスクトップクライアントで、〈新規チェックインの時、アーカイブを自動作成する〉オプションが設定されているとき。
PostCreateArchive	<p>新規のアーカイブを作成した後。次の場合には、当初のリビジョンをチェックインするときにもこのイベントが発生します。</p> <ul style="list-style-type: none"> ■ AutoCreate ディレクティブがコマンドラインインターフェイス中で有効になっているとき。 ■ デスクトップクライアントで、〈新規チェックインの時、アーカイブを自動作成する〉オプションが設定されているとき。

図 9-1 : デスクトップクライアントのイベント名



カスタムツールの実行

カスタムツールは、Version Manager デスクトップクライアントでのみ使用できます。Version Manager デスクトップクライアントでは、ツールバー上のアイコンまたは [ツール] メニューのメニュー項目、またはその両方として表示されるカスタムツールを追加できます。この機能は、頻繁に使用されるアプリケーションやタスクへのショートカットを作成する場合に便利です。

カスタムツールの定義方法の詳細については、[182 ページ](#)の『[カスタムツールの追加](#)』を参照してください。

イベントトリガのセットアップ

このセクションでは、デスクトップクライアントとコマンドラインインターフェイスでイベントトリガをセットアップする方法について説明します。

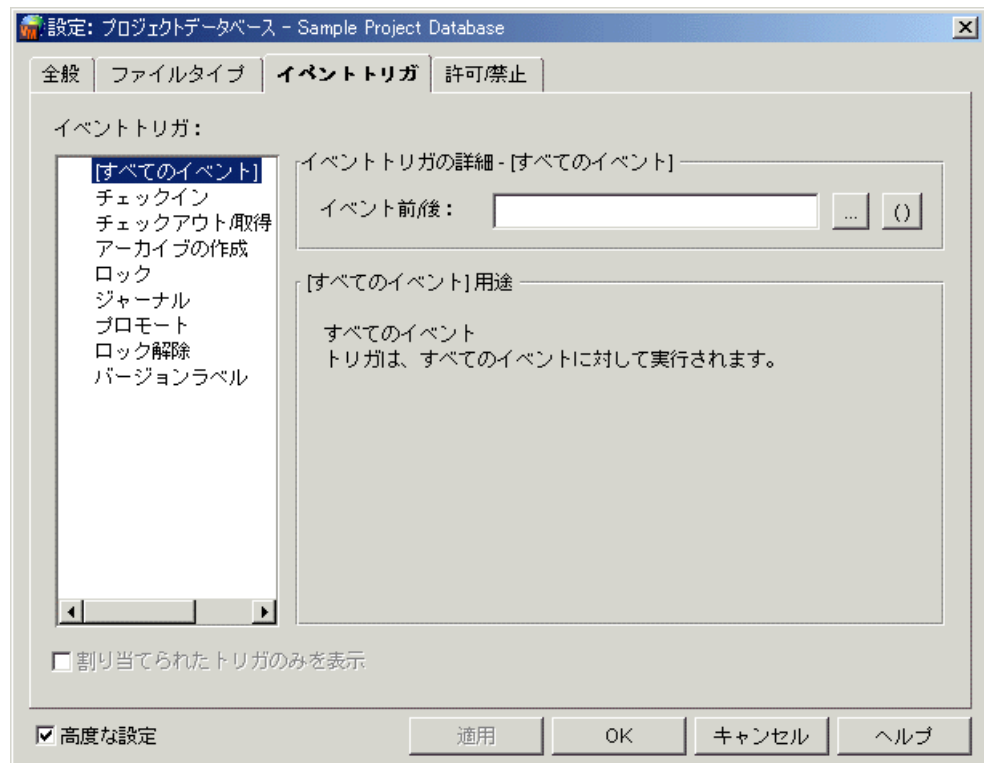
デスクトップクライアントを使用する場合

イベントトリガをセットアップするには、セットアップを行うユーザに「プロジェクトの設定」権限が割り当てられており、使用するプロジェクトデータベースに〈イベントトリガ〉コンフィグレーションオプションの使用が許可されていることが必要です。そうでない場合は、この手順は実行できません。

この手順では、イベントトリガファイルが定義されていることを前提としています。サポートされるイベントトリガファイルは [272 ページ](#)に記載されています。

イベントトリガをセットアップするには

- 1 Version Manager イベントの前または後に実行するイベントトリガを決定します。
- 2 イベントトリガを定義するプロジェクトデータベースまたはプロジェクトを選択します。
- 3 [管理] - [詳細設定] を選択して、[プロジェクトの詳細設定] ダイアログボックスを表示します。
- 4 [イベントトリガ] タブをクリックします。



- 5 左側の〈イベントトリガ〉リストから、イベントトリガを定義するイベントを選択します。
- 6 〈イベントトリガの詳細〉グループボックスにイベントトリガファイルのパスと名前を入力します。たとえば、Windows では「チェックイン」イベントの〈後処理〉テキストボックスに `C:\¥BIN¥POSTPUT.BAT` と入力します。また、イベントの情報をイベントトリガに渡すこともできます。この手順については [276 ページの『イベントトリガへのイベント情報の転送』](#) を参照してください。
- 7 さらにイベントトリガをセットアップするには、まず【適用】をクリックして現在のイベントトリガの定義を保存してから、手順 5 と 6 を繰り返します。それ以外の場合は【OK】をクリックします。

コマンドラインインターフェイスを使用する場合

Version Manager のコマンドラインインターフェイスを使用することが多い場合は、テキストエディタを使用してコンフィグレーションファイルを編集することもできます。または、デスクトップクライアントを使用して設定するほうが簡単な場合もあります。デスクトップクライアントを使用してイベントトリガ情報を変更したコンフィグレーションファイルは、コマンドラインインターフェイスでも有効です。

この手順では、イベントトリガファイルが定義されていることを前提としています。サポートされるイベントトリガファイルは [272 ページ](#)に記載されています。

イベントトリガをセットアップするには

- 1 Version Manager イベントの前または後に実行するイベントトリガを決定します。
- 2 テキストエディタを使用して、イベントトリガを追加するコンフィグレーションファイルを開きます。
- 3 イベントトリガを追加します。イベントトリガごとに、異なる行を使用します。

```
EventTrigger [=] event_name [event_trigger] [event_information]
```

ここで、

event_name は [272 ページの表 9-1 : 処理イベント](#)にある、いずれかのイベントです。

event_trigger はイベントトリガファイルのパスと名前を指定します。たとえば、Windows では C:¥BIN¥POSTPUT.BAT、UNIX では /usr/bin/postput のように入力します。

event_information については、[276 ページの『イベントトリガへのイベント情報の転送』](#)を参照してください。

次にその例を示します。

Windows EventTrigger = AllEvents X:¥PVCS¥EVENT¥ALL.BAT

UNIX EventTrigger = AllEvents /usr/pvcs/event/all

- 4 コンフィグレーションファイルを保存し、テキストエディタを終了します。

イベントトリガへのイベント情報の転送

デスクトップクライアントまたはコマンドラインインターフェイスでは、アクションの対象となっているアーカイブに関する情報をイベントトリガに転送できます。この情報はアーカイブに関する情報の収集や、イベントトリガが正しいアーカイブに対してアクションを行っていることの検証に使用されます。

デスクトップクライアントでは、アクションの対象となっているプロジェクトに関する情報もイベントトリガに送ることができます。

たとえば、誰かがリビジョンをチェックインするときには、電子メールによって通知してもらうことにします。この場合、電子メールによる通知メッセージを作成するプログラムを作成し、新しいリビジョンがチェックインされると、このプログラムが実行されるよう、Version Manager を設定します。作成したプログラムは、ユーザ ID、アーカイブと作業ファイルの名称、リビジョン番号、および時刻などの情報にアクセスできます。プログラムは、この情報をテキストファイルとしてフォーマットし、このファイルをメッセージとしてメールを送るよう、電子メール取込みプログラムに指示します。作成したプログラムはイベントトリガになります。イベントは PostPut です。

[表 9-2](#) はイベントトリガに得られる情報の一覧です。この情報の一部は、[279 ページの表 9-3 : イベントごとに得られるイベント情報](#)に示すように、特定のイベントからのみ得られます。

表 9-2 : イベント情報

イベント情報	定義
EventArchive	アーカイブの名前を含む、有効なフルパス。
EventArchiveName*	アーカイブの名前。
EventChgDesc	コメントの変更。パラメータファイルを使用してイベント情報を渡す場合以外は、1KB までにサイズが制限されています (284 ページの『パラメータファイルによる方法』参照)。
EventClientWorkfile*	クライアント上の作業ファイルの名前、あるいはスタンドアロンの場合には完全なパスを含む、有効なフルパス。
EventConfigFiles*	Version Manager が読み込む順にリストアップされたコンフィグレーションファイルの名前を含む完全修飾パス。このリストにはマスターコンフィグレーションファイルは含まれません。それぞれの値の間はセミコロン (;) で区切られます。
EventDate	イベントトリガが実行された日付。この日付のフォーマットは 10 進法による数値になります。例: 2003 年 1 月 20 日は 20030120 となります。
EventEntityPath*	イベントトリガまたはツールバーの操作対象のエンティティの PCLI 互換エンティティパス (¥<project>¥<subproject>¥...¥<file>)。
EventFolderPath*	5.3/6.0 フォルダ内のフォルダ、バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ。これは、5.3/6.0 フォルダにのみ適用され、プロジェクトには適用されません。
EventFQPWorkfile	作業ファイルの名前を含む、有効なフルパス。
EventGroup	リビジョンに割り当てられたプロモーショングループ。イベントが、プロモーショングループの割り当てまたは変更を行う操作の一部である場合は、新しいまたはプロモートされたグループの名前となります。そうでなく、プロモーショングループを使用してバージョン管理アイテムが選択された場合は、そのグループ名となります。
EventItemSelectionType*	処理対象の項目がどのように選択されたか。値は、ProjectDB、Project、Folder、File、Revision、VersionLabel、および Group (プロモーショングループ) のいずれかです。
EventJournalEntry	ジャーナルファイルに書き込まれた、ジャーナルエントリの行。
EventJournalFile	ジャーナルファイルの名前。
EventLock	ロックが要求され、それ以外に定義されていない場合は、値が YES になります。
EventMasterConfigFile*	マスターコンフィグレーションファイルの名前を含む完全修飾パス。
EventName	イベントの名前。アイコンおよびメニュー項目イベントに関しては、EventName の名前は常に ButtonEvent になります。
EventParmFile	イベントトリガ情報の全 <変数>=<値>ペアを含むテンポラリーファイルへのパス名。プログラムに渡される環境変数または引数を読む代わりに情報をファイルから得たいプログラムを対象としています。
EventPassword*	プロジェクトデータベースにログインするために使用するパスワードに基づいた、有効期限が定められた暗号化パスワード (24 時間有効)。ファイルサーバ使用時の認証に使用されます。

イベント情報	定義
EventProjectDB*	使用中のプロジェクトデータベースの名前を含む完全修飾パス。
EventProjectName*	プロジェクトデータベース中のプロジェクトの名前を含む完全修飾パス。
EventRevision	作業中のリビジョンの番号。
EventTime	イベントトリガが実行された時刻。この時刻のフォーマットは 10 進法による数値になります。たとえば午後 2:35:26 は 143526 となります。
EventUserID	作業を行っているユーザのユーザ ID。
EventVersion	バージョンラベル。イベントが、バージョンラベルの割り当てまたは変更を行う操作の一部である場合は、新しいラベルとなります。そうでなく、バージョンラベルを使用してバージョン管理アイテムが選択された場合は、そのラベル名となります。
EventWorkfile	作業ファイルの名前。

* CLI コマンドから実行するトリガでは取得できません。

表 9-3 : イベントごとに得られるイベント情報

VM イベント	AllEvents	PreGet	PostGet	Unconditional PrePut	PrePut	PostPut	PostJournal	PrePromote	PostPromote	PreVersionLabel	PostVersionLabel	PreLock	PostLock	PreUnlock	PostUnlock	PreCreateArchive	PostCreateArchive	Icon/Menu
EventArchive	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	□
EventArchiveName*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	□
EventChgDesc	□				□	□												
EventClientWorkfile*	□	■	■	■	■	■	□	■	■	■	■	■	■	■				□
EventConfigFiles*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	□
EventDate	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventEntityPath*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventFolderPath*	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
EventFQPWorkfile	□	■	■	■	■	■												
EventGroup	□	□	□	□	□	□	□	■	■	□	□	□	□	□				□
EventItemSelectionType*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventJournalEntry	□	□	□	□			■											
EventJournalFile	□						■											
EventLock	□	□	□	□	□	□												
EventMasterConfigFile*	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□
EventName	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventParmFile	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventPassword*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventProjectDB*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventProjectName*	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	□
EventRevision	□	■	■	□	■	■	□			■	■	■	■	■				□
EventTime	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventUserID	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
EventVersion	□	□	□	□	□	□	□			■	■	□	□	□	□			□
EventWorkfile	□	■	■	■	■	■				■	■							□

- このイベント情報は、常に取得できます。
 - このイベント情報は、取得できる場合とできない場合があります。たとえば、グループ情報 (EventGroup) は PrePromote や PostPromote に対しては常に取得できますが、Get や Put では指定または選択されている場合にのみ取得できます。
- *CLI コマンドから実行するトリガでは取得できません。

カスタムツールのイベント情報

カスタムツールに渡すことができるイベント情報は、そのイベントが発生したときにデスクトップクライアントで選択されている項目に依存します。たとえば、EventArchive イベント情報は、イベントトリガに対し、アイコンおよびメニュー項目イベントが発生したときにデスクトップクライアント中でバージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループが選択されていた場合に得られます。

次の表には、カスタムツールにイベント情報を渡すために、デスクトップクライアント中で選択されていない項目がイベント情報ごとに記載されています。項目が選択されていない場合は、Version Manager は対応する情報をカスタムツールに渡しません。

たとえば、作業ファイルの名前（EventWorkfile）を要求するカスタムツールが定義され、そのイベントトリガの実行時にバージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループが選択されていない場合は、その作業ファイルの名前はカスタムツールには渡されません。

イベント情報	デスクトップクライアント中で選択されていない項目
EventArchive	バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ
EventArchiveName	バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ
EventWorkfile	バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ
EventUserID	プロジェクトデータベース以下のレベルのアイテム
EventRevision	リビジョン
EventVersion	バージョンラベル
EventGroup	プロモーショングループ
EventDate	プロジェクトデータベース以下のレベルのアイテム
EventTime	プロジェクトデータベース以下のレベルのアイテム
EventConfigFiles	プロジェクトデータベース以下のレベルのアイテム
EventMasterConfigFile	プロジェクトデータベース以下のレベルのアイテム
EventClientWorkfile	バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ
EventProjectDB	プロジェクトデータベース以下のレベルのアイテム
EventProjectName	プロジェクト以下のレベルのアイテム
EventFolderPath	5.3/6.0 フォルダ内のフォルダ、バージョン管理ファイル、リビジョン、バージョンラベル、またはプロモーショングループ。これは、5.3/6.0 フォルダにのみ適用され、プロジェクトには適用されません。
EventItemSelectionType	プロジェクトデータベース以下のレベルのアイテム
EventName	プロジェクトデータベース以下のレベルのアイテム。 EventName に対するアイコンまたはメニュー項目イベントの値は常に ButtonEvent になります。

イベント情報を渡す方法と処理の中断

Version Manager は、以下を使用してイベント情報を渡します。

- 環境変数
- コマンドラインマクロ
- パラメータファイル

イベント情報が環境変数を経由して渡される場合は、イベントトリガは必要とする値を環境変数から取得します。環境変数を使用してイベント情報を渡せない場合は、コマンドラインマクロまたはパラメータファイルを使用します。

すべてのイベントトリガが、イベント情報を渡す方法をどれでも使用できるわけではありません。また、処理の中断が可能なイベントと不可能なイベントがあります。名前が「Pre」で始まる（前処理）イベントは処理の中断が可能ですが、「Post」で始まる（後処理）イベントには処理を中断できないものがあります。たとえば、チェックインするプログラムが構文エラーや警告を受けずに正しくコンパイルされていることを確認するため、PrePut イベントトリガを作成したとします。コンパイルに失敗した場合は、このイベントトリガがゼロ以外の終了コードを戻して、チェックインを中断するよう Version Manager に伝えることができます。この場合、Version Manager はチェックイン (Put) を中断します。

環境変数の使用

イベント情報が環境変数を経由して渡される場合は、イベントトリガは必要とする値を環境変数から取得します。たとえば、バッチファイルは %EVENTARCHIVE% を、UNIX シェルスクリプトは \$EVENTARCHIVE を、ビルドスクリプトは EVENTARCHIVE を参照します。また、C プログラムは getenv("EVENTARCHIVE") 関数呼び出しを行います。

Windows 以下は、POSTPUT.BAT という名前のバッチファイルの例です。

```
H:¥PUBLIC¥SEND "Just checked in %EVENTWORKFILE%, revision
    %EVENTREVISION% to group %EVENTGROUP%"
```

イベントトリガ、

```
POSTPUT C:¥BIN¥POSTPUT.BAT
```

は、Version Manager による作業ファイルのチェックインが成功した後に、作業グループにメッセージを送ります。

UNIX 以下は、UNIX postput シェルファイルの例です。

```
/usr/pvcstools/send "Just checked in $EVENTWORKFILE, revision
    $EVENTREVISION to group $EVENTGROUP"
```

イベントトリガ、

```
POSTPUT /usr/bin/postput
```

は、Version Manager による作業ファイルのチェックインが成功した後に、作業グループにメッセージを送ります。

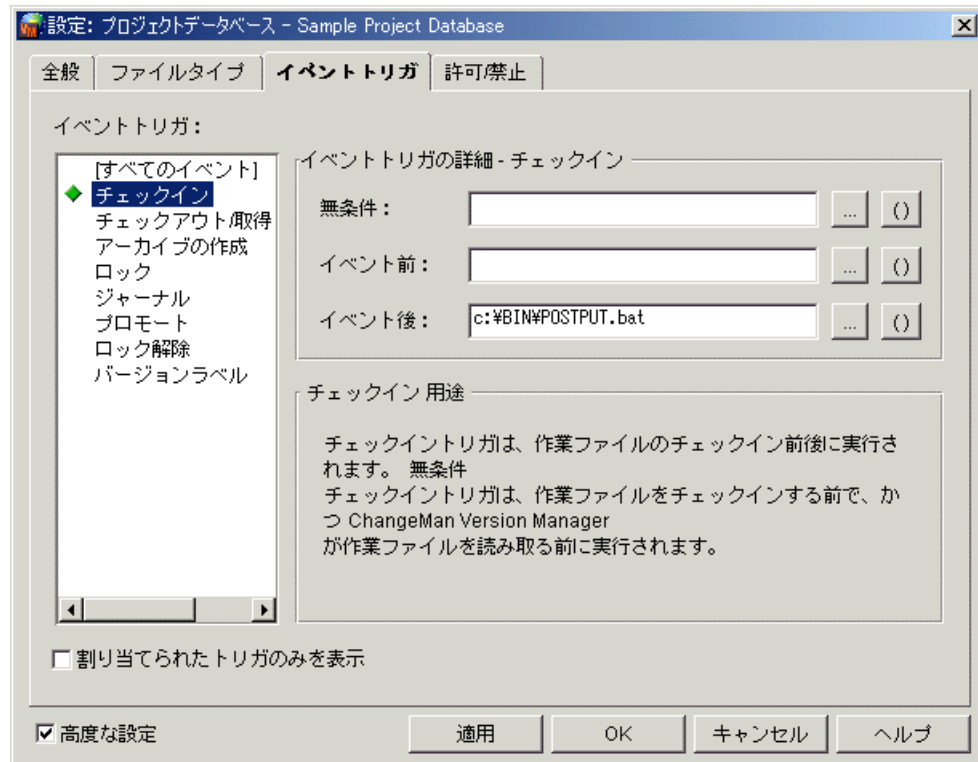
デスクトップクライアントでの指定

[プロジェクトの詳細設定] ダイアログボックスの〈イベントトリガの詳細〉グループボックス ([管理] - [詳細設定] の [イベントトリガ] タブ) に、イベントトリガファイルのパスと名前を入力し

ます。環境変数はイベントトリガファイル中で指定されます。たとえば、「チェックイン」イベントを選択し、次のように〈後処理〉テキストボックスに入力します。

Windows C:¥BIN¥POSTPUT.BAT

UNIX /usr/bin/postput



コマンドラインインターフェイスでの指定

テキストエディタを使用して、該当するコンフィグレーションファイルにイベントトリガの定義を追加します。環境変数はイベントトリガファイル中で指定されます。次に例を示します。

```
EventTrigger = POSTPUT C:¥BIN¥POSTPUT.BAT
```

コマンドラインマクロの使用

イベントトリガの定義は、(イベントトリガファイル中ではなく) コマンドライン上で指定されたマクロを経由してイベント情報を参照する場合があります。これらのコマンドラインマクロの名前は、[277 ページの表 9-2: イベント情報](#)にあるイベント情報の名前と同じになります。この名前を指定するには、たとえば `__EVENTARCHIVE__` のように名前を 2 個のアンダースコア (`__`) で囲みます。

次に示す例では、Configuration Builder を呼び出し、アーカイブの名前を含む ARCHIVE というマクロを定義するイベントトリガが定義されています。

```
EventTrigger = PREPUT Build.exe -FMYEVENT.BLD ARCHIVE=__EVENTARCHIVE__
```

Version Manager が PrePut イベントを実行するときに、このコマンドライン実行ファイルは次のように展開されます。

```
Build.exe -FMYEVENT.BLD ARCHIVE=K:¥SOURCE¥PVCS¥LOGFILES¥GET.C-ARC
```

UNIX コマンドラインインターフェイスを使用する場合：展開後にスペースなどの特殊文字が含まれる可能性のあるコマンドラインマクロの場合はすべて、マクロ名を引用符（'または"）で囲む必要があります。たとえば、コマンドラインマクロによる方法（__EventJournalEntry__）を使用して PostJournal イベントに EventJournalEntry 情報を渡す場合（__EventJournalEntry__）は、このマクロの名前を引用符で囲まなければなりません（'__EventJournalEntry__'）。これは、EventJournalEntry によって渡される情報にはかっこを含めることができませんが、かっこが UNIX コマンドラインでは特殊文字であるためです。

ただし、以下のマクロは、引用符で囲む必要はありません。

- __EventName__
- __EventRevision__
- __EventDate__
- __EventTime__
- __EventParmFile__

デスクトップクライアントでの指定

処理イベントの場合

〔プロジェクトの詳細設定〕ダイアログボックスの〈イベントトリガの詳細〉グループボックス（〔管理〕－〔詳細設定〕の〔イベントトリガ〕タブ）に、イベント情報を渡すコマンドラインマクロを含めてイベントトリガの定義を入力します。たとえば、「チェックイン」イベントを選択し、次のように〈前処理〉テキストボックスに入力します。

```
Build.exe -FMYEVENT.BLD ARCHIVE=__EVENTARCHIVE__
```

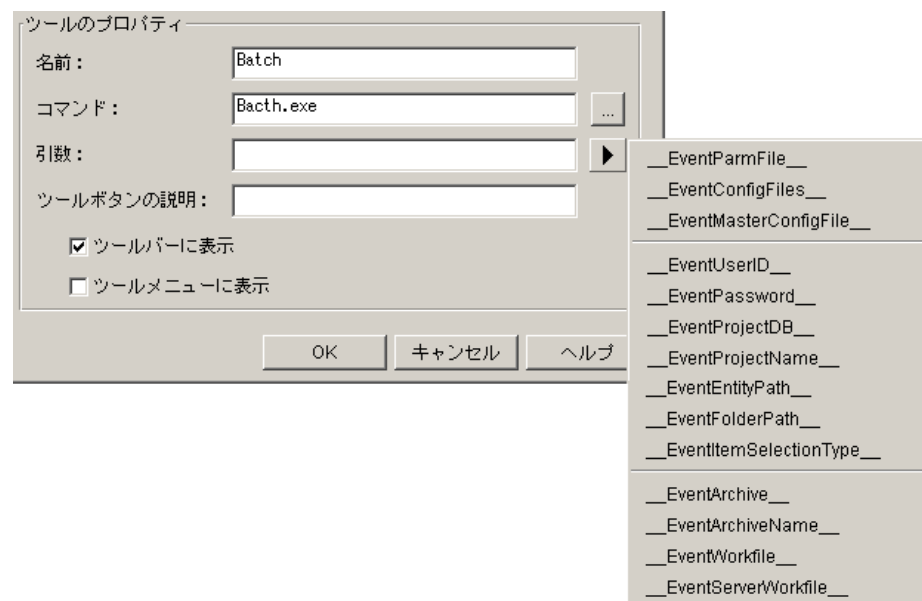


注 コマンドラインマクロを渡すには、〈前処理〉、〈後処理〉、または〈無条件〉テキストボックスの隣にある【()】をクリックし、マクロを選択します。

アイコンおよびメニュー項目イベントの場合

〔ツールの設定〕ダイアログボックスの〈引数〉テキストボックス（〔管理〕－〔ツールの設定〕）で、〈引数〉テキストボックスの隣にあるボタンをクリックして実行ファイルにイベント情報を渡すコマンドラインマクロを選択します。

次にその例を示します。



コマンドラインインターフェイスでの指定

テキストエディタを使用し、イベント情報を渡すコマンドラインマクロを含めたイベントトリガの定義を、該当するコンフィグレーションファイルに追加します。

Windows の場合 :

```
EventTrigger = POSTPUT = E:¥WD¥POST.BAT __EVENTFQPWORKFILE__
```

UNIX の場合 :

```
EventTrigger = POSTPUT = /usr/wd/post __EVENTFQPWORKFILE__
```

パラメータファイルによる方法

イベント情報を渡すためにパラメータファイルを使用する場合は、(イベントトリガファイル中ではなく) コマンドライン上でマクロ、__EventParmFile__ を指定します。このパラメータファイルによる方法はもっとも遅いため、特定のイベントトリガに関して環境変数による方法が使用できない場合、または変更コメントが 1K よりも大きい場合にのみ使用してください。

この方法を使用すると、実行されるイベントトリガが利用できるイベント情報の値を含むテンポラリファイルが、Version Manager によって作成されます。イベントトリガは次に、この情報を使用します。すべての Version Manager イベントがすべてのイベント情報を利用できるわけではありません。

次の例では、__EventParmFile__ の使用方法を示しています。この例のイベントトリガの定義では、Configuration Builder を呼び出し、現在の作業ディレクトリにテンポラリファイルを作成します。Version Manager は各マクロ定義の名前と値を、*name=value* の構文に従って記述します。

```
EventTrigger = ALLEVENTS Build.exe -FMYEVENT.BLD EventInfoFile=__EventParmFile__
```

この例では、Version Manager がいずれかの Version Manager イベントを実行するときに、コマンドライン実行ファイルが次のように展開されます。

```
Build.exe -FMYEVENT.BLD EventInfoFile=E:¥TMP¥PVCS0024.TMP
```

Version Manager はテンポラリファイルを作成し、イベントトリガを実行後、このファイルを削除します。テンポラリファイルは EventInfoFile で指定され、その内容が Configuration Builder で使用されます。

ここで、たとえばファイル、E:¥TMP¥PVCS0024.TMP の内容は次のようになります。

```
EVENTNAME=PostPut
EVENTARCHIVE=E:¥TMP¥ET¥FOO.C_V
EVENTWORKFILE=FOO.C
USERID=BLAIR
EVENTLOCK=
EVENTREVISION=1.1
EVENTVERSION=
EVENTGROUP=
EVENTJOURNALFILE=
EVENTJOURNALENTY=
EVENTDATE=19990123
EVENTTIME=112104
EVENTFQPWORKFILE=D:¥TMP¥ET¥FOO.C
EVENTCONFIGFILES=
EVENTMASTERCONFIGFILE=E:¥PRJDB¥ARCHIVES¥CPE2QP.CFG
EVENTARCHIVEPATH=
EVENTCLIENTWORKFILE=
```

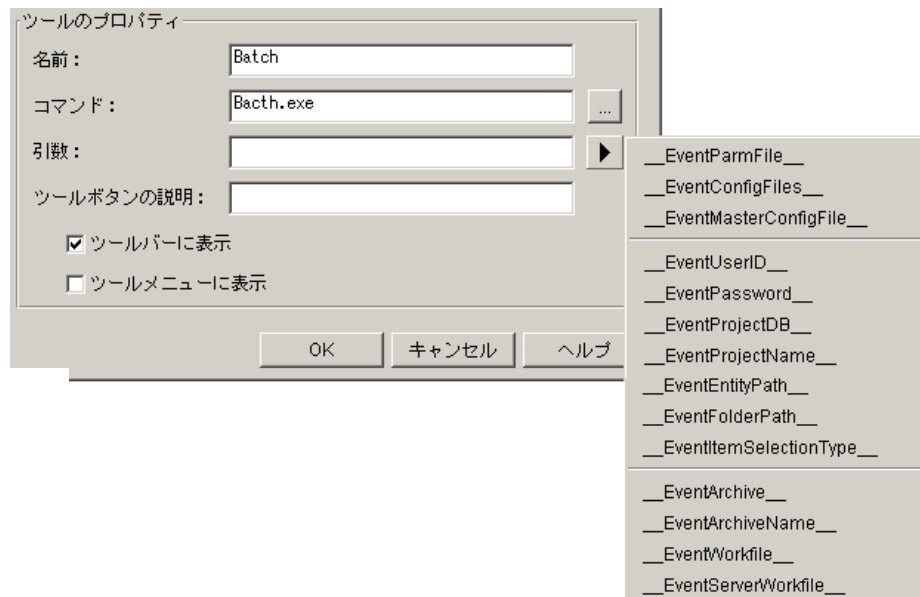
```
EVENTPROJECTDB=
EVENTPROJECTNAME=
EVENTITEMSELECTIONTYPE=
EVENTFOLDERPATH=
EVENTCHGDESC=No change.
```

デスクトップクライアントでの指定

処理イベントの場合 [プロジェクトの詳細設定] ダイアログボックスの〈イベントトリガの詳細〉グループボックス ([管理] - [詳細設定] の [イベントトリガ] タブ) に、パラメータファイルとしてのイベント情報を含めてイベントトリガの定義を入力します。たとえば「[すべてのイベント]」イベントを選択し、次のように〈イベント前/後〉テキストボックスに入力します。

```
Build.exe -FMYEVENT.BLD EventInfoFile=__EventParmFile__
```

アイコンおよびメニュー項目イベントの場合 [ツールの設定] ダイアログボックスの〈引数〉テキストボックス ([管理] - [ツールの設定]) で、〈引数〉テキストボックスの隣にあるボタンをクリックして表示されるドロップダウンリストから「__EventParmFile__」を選択します。



コマンドラインインターフェイスでの指定

テキストエディタを使用し、パラメータファイルとしてのイベント情報を含めたイベントトリガの定義を、該当するコンフィグレーションファイルに追加します。次にその例を示します。

Windows の場合 :

```
EventTrigger = ALLEVENTS E:¥PVCS¥EVENT¥ALL.EXE EventInfoFile=__EventParmFile__
```

UNIX の場合 :

```
EventTrigger = ALLEVENTS /usr/pvcs/event/all EventInfoFile=__EventParmFile__
```

イベントトリガの例

例 1 : チェックインされるリビジョンに浮動ラベルを付けるための、PostPut イベントトリガを作成します。

次のイベントトリガは、ブランチを含めてチェックインされるすべてのリビジョンに浮動ラベル、LATEST を配置します。浮動ラベルをトランクリビジョンのみに付ける場合にはイベントトリガは不要です。その代わりにプロジェクト中のすべてのアーカイブのトランクに浮動ラベルを付けることができます。コンフィグレーションファイル中のイベントトリガの定義は次のとおりです。

```
EventTrigger = POSTPUT vcs -vLATEST: __EventRevision__ * -y __EventArchive__
```

このイベントトリガには、コマンドラインマクロを経由してイベント情報が渡されます。

例 2 : プロモートされるリビジョンをロックせずにチェックアウト (get) し、それを指定されたディレクトリに配置するための PrePromote イベントトリガを作成します。

次のイベントトリガは、プロモートされるリビジョンをロックせずにチェックアウト (get) し、それが Stage プロモーショングループにプロモートされる場合と Production プロモーショングループにプロモートされる場合に応じて、それぞれのディレクトリに配置します。コンフィグレーションファイル中のこのイベントトリガの定義は次のとおりです。

```
EventTrigger = [path]prepromote.bat __EventGroup__ __EventArchive__
```

prepromote.bat ファイルの内容は次のようになります。

```
@echo off
rem %1 = EventGroup and %2 = EventArchive
IF "%1" == "Development" GOTO STAGE
IF "%1" == "Stage" GOTO PRODUCTION
:STAGE
get -cD:%PVCSPROJ%prmsmple.prj%vcs.cfg
    -gDevelopment %2(c:%staging%famis)
    :PRODUCTION
get -cD:%PVCSPROJ%prmsmpl.prj%vcs.cfg
    -gStage %2(c:%production%famis)
```

例 3 : すべてのイベントに渡されるイベント情報をエコーする「[すべてのイベント]」イベントトリガを作成します。

次のイベントトリガは、それぞれの Version Manager イベントで得られるイベント情報をコンソールウィンドウに表示します。このイベントトリガは、どのイベントに伴ってどの情報が渡されるのかをチェックする場合に便利です。コンフィグレーションファイル中のイベントトリガの定義は次のとおりです。

```
EventTrigger = [path]echovm.bat
```

echovm.bat ファイルは、イベント情報を得るために環境変数を使用します。このファイルの内容は次のとおりです。

```
echo off
echo Action %EventName% on %EventDate% %EventTime%
echo.
echo EventName:%EventName%
echo EventArchive:%EventArchive%
echo EventWorkfile:%EventWorkfile%
echo EventUserID:%EventUserID%
echo EventLock:%EventLock%
echo EventRevision:%EventRevision%
```

```
echo EventVersion:%EventVersion%  
echo EventGroup:%EventGroup%
```


第 10 章

レポートの使用

はじめに	290
レポートオプションの設定	290
HTML レポートのフォーマットのカスタマイズ	292
ジャーナルレポートの生成	293
履歴レポートの生成	297
セキュリティレポートの生成	302
プロジェクトに対する変更の表示 (change.log)	304

はじめに

Serena ChangeMan Version Manager では、以下の 3 種類のレポートを作成できます。

- ジャーナルレポート。アーカイブを更新するアクションに関する情報が含まれます。
- 履歴レポート。アーカイブと作業ファイルの名前、アーカイブの記述、リビジョン番号、リビジョン履歴などのアーカイブに関する情報が含まれます。履歴レポートは以前のバージョンの Version Manager ではアーカイブレポートと呼ばれていました。
- セキュリティレポート。アクセスコントロールデータベース中のセキュリティ設定が含まれます。

デスクトップクライアントでは、レポートを HTML ブラウザまたはテキストエディタにより表示できます。Windows プラットフォームで HTML フォーマットによる表示を行うと、デフォルトのブラウザが使用されます。UNIX プラットフォームでレポートを HTML フォーマットにより表示する場合は、使用する HTML ブラウザを指定する必要があります。HTML レポートのフォーマットはカスタマイズ可能です。



注 400 個以上のファイルに対して生成された履歴レポートを HTML ブラウザを使用して表示すると、表示に 1、2 分かかります。より高速に表示するにはテキストエディタを使用してください。

この章では、レポートオプションの設定、HTML レポートフォーマットのカスタマイズおよび、デスクトップクライアントまたはコマンドラインインターフェイスによりレポートを生成する手順について説明します。

変更ログ

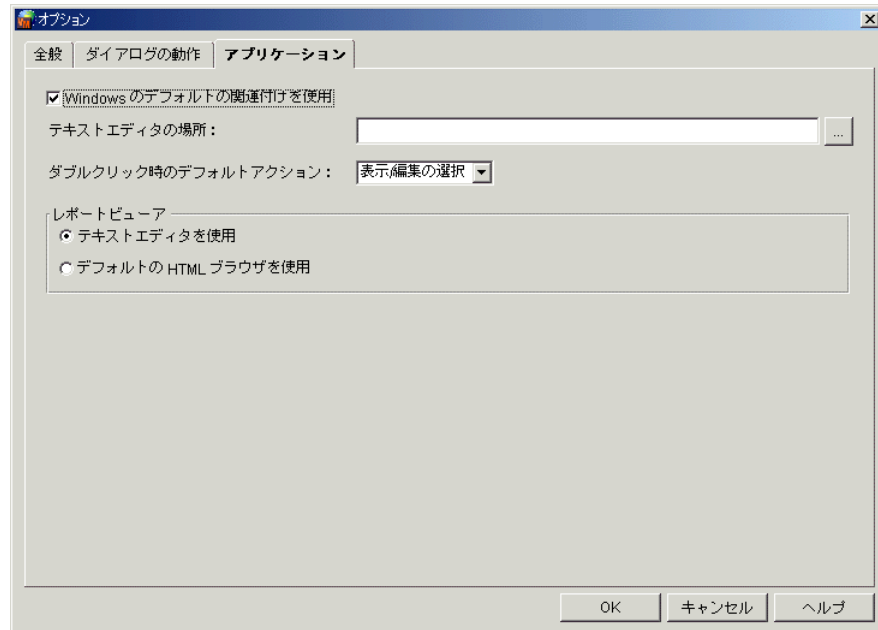
これらのレポートに加え、Version Manager では、サブプロジェクトやバージョン管理ファイルの追加や削除など、プロジェクトの内容に対する変更も記録されます。これらの変更は、プロジェクトデータベースのルートディレクトリにある `change.log` ファイルに記録されます。詳細については、[304 ページの『プロジェクトに対する変更の表示 \(change.log\)』](#)を参照してください。

レポートオプションの設定

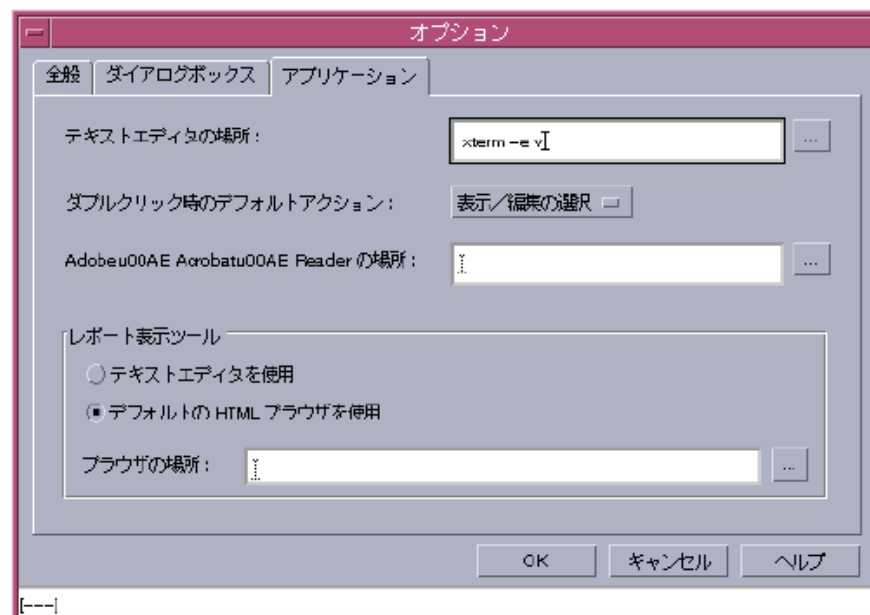
レポートオプションを設定するには

- 1 [表示] - [オプション] を選択します。[オプション] ダイアログボックスが開き、[全般] タブがアクティブになっています。
- 2 [アプリケーション] タブをクリックします。〈レポート表示ツール〉グループボックスから、次のいずれかを選択します。
 - テキストエディタを使用

- デフォルトの HTML ブラウザを使用



注 UNIX 版で、〈デフォルトの HTML ブラウザを使用〉を選択している場合は、〈ブラウザの場所〉テキストボックスにブラウザの場所を入力する必要があります。



HTML レポートのフォーマットのカスタマイズ

Version Manager がジャーナル、履歴、およびセキュリティレポートの表示に使用する HTML テンプレートはカスタマイズが可能です。レポートのタイプごとに、`journal.template`、`history.template`、および `accessdb.template` のそれぞれテンプレートが存在します。デフォルトでは、これらのファイルは Version Manager とともに次の場所にインストールされています。

- Windows の場合 : `drive:¥Program Files¥Serena¥vm¥common¥pvcsprop¥pvcs¥vm`
- UNIX の場合 : `/usr/serena/vm/common/pvcsprop/pvcs/vm`

これら 3 つのデフォルトテンプレートは同一で、内容は次のとおりです。

```
<HTML>
<HEAD>
<TITLE><subst data="REPORT_TITLE_PROPERTY"></subst></TITLE>
<META NAME=GENERATOR CONTENT="Version Manager">
</HEAD>
<BODY BGCOLOR="#FFFFFF" LINK="#0000EE" VLINK="#0000EE" ALINK="#9933EE"
      TEXT="#000000" LEFTMARGIN="8" TOPMARGIN="8" BACKGROUND="/
      vminet_images/Bkgrd.gif">
<TABLE BORDER="0" CELSPACING="0" CELLPADDING="0">
<TR VALIGN="top">
<TD>
<B><FONT SIZE="+2"><subst data="REPORT_TITLE_PROPERTY"></subst></
      FONT></B>
</TD>
</TR>
<TR VALIGN="top">
<TD>
<HR WIDTH="100%" SIZE="2" COLOR="Black" NOSHADE>
</TD>
</TR>
<TR VALIGN="top">
<TD>
<PRE>
<subst data="REPORT_FILE_PROPERTY"></subst>
</PRE>
</TD>
</TR>
<TR VALIGN="top">
<TD>
<HR WIDTH="100%" SIZE="2" COLOR="Black" NOSHADE>
</TD>
</TR>
</TABLE><!--Footnote-->
<P>
</BODY>
</HTML>
```

レポートのタイトルを表示するためのタグは、以下のとおりです。

```
<subst data="REPORT_TITLE_PROPERTY"></subst>
```

レポートを表示するためのタグは、以下のとおりです。

```
<subst data="REPORT_FILE_PROPERTY"></subst>
```

テンプレートファイルの書式タグはどれでも変更できます。たとえば、フォントの色やサイズ、背景色を変更することによってレポートをカスタマイズできます。書式タグ、テキスト、ほかのページへのリンク、グラフィックファイル（企業のロゴなど）などを追加することもできます。

ジャーナルレポートの生成

ジャーナルレポートには、バージョンラベルの割り当て、作業ファイルのチェックインとチェックアウト、プロモーショングループの割り当てなど、アーカイブに対して加えられた変更に関する情報が含まれます。

ジャーナルレポートは、ジャーナルファイル中の情報に基づいて生成されます。ジャーナルファイルには、ユーザがアーカイブに対して行うアクションの記録が含まれます。Version Manager がジャーナルファイルを保持するように設定されていない場合は、ジャーナルレポートは生成できません。ジャーナルファイルを保持するように設定する方法については、[89 ページの『ジャーナルファイルのオプション』](#)を参照してください。

ジャーナルレポートは、あるアーカイブについていつリビジョンがチェックアウトされたか、あるいはいつバージョンラベルが割り当てられたかなど、アーカイブへの作業に関する特定の情報を得るときに便利です。

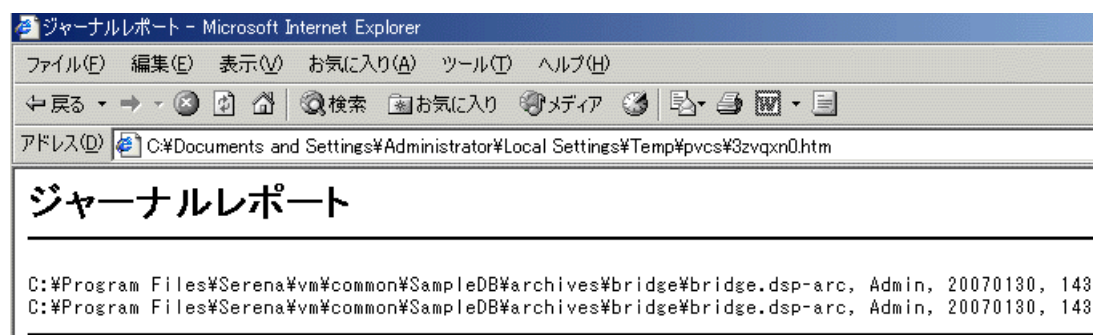
ジャーナルレポートのデータはすでにジャーナルファイル中にあり、生成するときにアーカイブを開く必要がないため、履歴レポートよりも速く生成できます。作成の日付やリビジョン情報、あるいはアーカイブへのアクセス制限など、アーカイブに関する一般的な情報が必要な場合には履歴レポートを生成します。[297 ページの『履歴レポートの生成』](#)を参照してください。

ジャーナルレポートを生成する前に、レポートの内容を制限するオプションを設定できます。オプションを設定したレポートでは、次のような表示を行えます。

- 特定の日付と日付の間における変更
- 特定のアーカイブに対して行われた変更
- 特定のアクション（コマンド）によるアーカイブへの作業
- 特定のユーザによって行われた変更
- 現在、特定のユーザによってロックされているリビジョン

ジャーナルレポートの読み方

デスクトップクライアントで生成され、HTML によって表示されるジャーナルレポートの例を次に示します。



ジャーナルレポートのフォーマットは、次のとおりです。

archive, user_id, date, time, action

ここで、

- *archive* はアーカイブのパスとファイル名です。
- *user_id* はアーカイブに変更を加えたユーザのユーザ ID です。
- *date* は変更の日付を十進数で表したものです。たとえば、2003 年 3 月 31 日は 20030331 と表されます。
- *time* は変更の時刻を 24 時間表示の十進数で表したものです。たとえば、午後 4:55:20 は 165520 と表されます。
- *action* は実行されたアクションのコマンドラインオプションです。たとえば、put -r1.1 はリビジョン 1.1 がチェックインされたことを示します。

デスクトップクライアントを使用する場合

デスクトップクライアントでは、プロジェクトデータベース、プロジェクト、複数のバージョン管理ファイル、または 1 つのバージョン管理ファイルについてジャーナルレポートを生成できます。プロジェクトデータベースまたはプロジェクトを選択した場合には、次のいずれかについてレポートを生成できます。

- プロジェクトデータベースまたはプロジェクト中の、すべてのプロジェクト / サブプロジェクトのすべてのバージョン管理ファイル
- プロジェクト / サブプロジェクトを含まず、選択されたプロジェクトデータベースまたはプロジェクト中のバージョン管理ファイルのみ

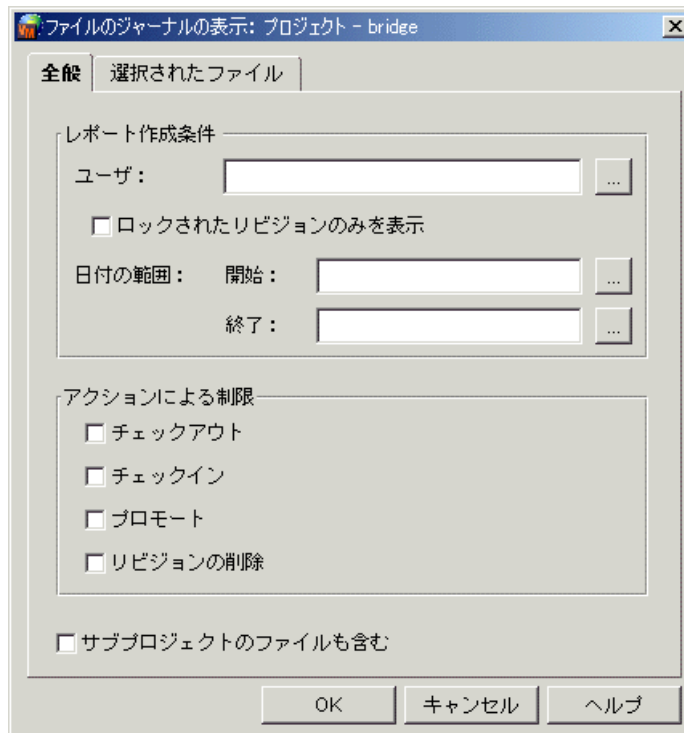
レポートの生成に使用されるジャーナルファイルは、選択した項目に関連付けられたコンフィグレーションファイルによって指定されます。選択された項目についてコンフィグレーションファイル中でジャーナルファイルが指定されていない場合には、ジャーナルレポートは生成できません。

ジャーナルレポートを生成するには、「ジャーナルの表示」権限が必要です。

デスクトップクライアントでジャーナルレポートを生成するには

- 1 プロジェクトデータベース、プロジェクト、またはバージョン管理ファイルを選択します。

- 2 [アクション] - [ジャーナルの表示] を選択します。[ジャーナルの表示] ダイアログボックスが開き、[全般] タブがアクティブになっています。



注 ファイル表示部でバージョン管理ファイルが選択されている場合は、〈サブプロジェクトのファイルも含む〉チェックボックスは表示されません。

- 3 [OK] をクリックすると、すべてのユーザによってアーカイブに加えられたすべての変更と、現在ロックされているすべてのリビジョンを含むレポートが生成されます。あるいは、次の作業を行って、レポートの内容を制限します。

レポート内容	操作
特定のユーザによって行われた変更	〈ユーザ〉テキストボックスに指定するユーザ ID を入力します。複数の ID はコンマで区切ります。
現在ロックされているリビジョン	〈ロックされたリビジョンのみを表示〉チェックボックスを選択します。〈ユーザ〉テキストボックスにユーザ ID を入力したうえで、このチェックボックスを選択した場合は、指定したユーザによって現在ロックされているリビジョンのみがレポートされます。
一定の期間に加えられた変更	期間の最初の日を〈開始日〉フィールドに、最後の日を〈終了日〉フィールドに入力するか、これらのフィールドの隣の【...】をクリックして、日付を選択します。日付と時間を、オペレーティングシステムで使用されている表示形式で入力する必要があります。 Windows では、[コントロールパネル] の [地域] で日付と時刻のフォーマットを設定します。UNIX では、PVCS_DATE_FORMAT および PVCS_TIME_FORMAT 環境変数により設定されます。

レポート内容	操作
一定の期間に加えられた変更 (続き)	コントロールパネルでの設定を行わない場合、あるいは環境変数が設定されていない場合は、mm/dd/yy hh:mm というフォーマットになります。 注 UNIX 上では、PVCS_DATE_FORMAT を MM/dd/yyyy (日を先にする場合は dd/MM/yyyy) に設定することをお勧めします。ここで、d と y はそれぞれ、日と年を表す数字です。M は月を表す英語の略号です。年は 4 桁で、月は大文字で指定してください。このフォーマットを使用すると、最適な結果が得られます。
リビジョンがチェックアウトされたアーカイブのみ	〈チェックアウト〉 チェックボックスを選択します。
リビジョンがチェックインされたアーカイブのみ	〈チェックイン〉 チェックボックスを選択します。
リビジョンがプロモートされたアーカイブのみ	〈プロモート〉 チェックボックスを選択します。
リビジョンが削除されたアーカイブのみ	〈リビジョンの削除〉 チェックボックスを選択します。
サブプロジェクト内のすべてのバージョン管理ファイルを含む、プロジェクト全体に対するレポート	〈サブプロジェクトのファイルも含む〉 チェックボックスを選択します。

- ジャーナルレポートを生成する前に選択されたバージョン管理ファイルを表示するには、[選択されたファイル] タブをクリックします。
- 【OK】をクリックします。ジャーナルレポートが生成されて表示されます。

コマンドラインインターフェイスを使用する場合

ジャーナルレポートを生成するには、JournalReport 権限が必要です。

コマンドラインインターフェイスを使用してジャーナルレポートを生成するには、VJOURNAL コマンドを使用します。このコマンドの構文は次のとおりです。

```
vjournal [option...] [journal_file...]
```

ここで、*journal_file* はレポートのための情報を得るジャーナルファイルです。ジャーナルファイルを指定しない場合は、コマンドラインで指定したコンフィグレーションファイル、または Version Manager に埋め込まれたマスターコンフィグレーションファイル中の Journal ディレクティブで指定されたジャーナルファイルが使用されます。Journal ディレクティブがファイルを指定していない場合は、VCSDir ディレクティブによって指定されたディレクトリが検索され、見つかったジャーナルファイルがすべてレポートの生成に使用されます。この VCSDir ディレクティブによりアーカイブディレクトリが指定され、デフォルトではこのディレクトリにジャーナルファイルが置かれます。

このコマンドでよく使用されるオプションを次の表に示します。

レポート内容	使用するコマンドライン
特定の日付と日付の間における変更	vjournal -ddate_range
特定のアーカイブに対して行われた変更	vjournal -larchive[,archive]

レポート内容	使用するコマンドライン
特定のコマンドによる作業	<code>vjournal -ocommand[,command]</code>
特定のユーザによって行われた変更	<code>vjournal -uuser_id[,user_id]</code>
現在ロックされているリビジョン	<code>vjournal -xl</code>

VJOURNAL コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

履歴レポートの生成

履歴レポートには、アーカイブまたはリビジョン、またはその両方に関する情報の概要が記載されません。これは、開発プロセスの監視、アーカイブ履歴の検証、およびアーカイブ属性のチェックなどに使用できる、アーカイブに関する情報を提供します。

履歴レポートは、以前のバージョンの Version Manager ではアーカイブレポートと呼ばれていました。

履歴レポートには、アーカイブに関する情報とリビジョンに関する情報が含まれます。アーカイブに関する情報には次のようなものがあります。

- アーカイブの作成日時と作成者
- アーカイブ属性
- リビジョンをロックしたユーザ
- アーカイブと作業ファイルの名前

リビジョンに関する情報には次のものがあります。

- リビジョンの記述
- リビジョンの履歴

履歴レポートを生成する前には、オプションを設定してレポートの内容を制限できます。レポートの内容は、たとえば次のように設定できます。

- アーカイブに関する情報は表示するが、リビジョンに関する情報は表示しない。
- リビジョンに関する情報は表示するが、アーカイブに関する情報は表示しない。
- 現在ロックされているリビジョン。
- 指定されたバージョンラベルに対応するリビジョン。
- 指定されたプロモーショングループに関連付けられたリビジョン。
- トランク上の最新のリビジョン。
- チップリビジョンが、指定されたリビジョン番号またはバージョンラベルと同一でないアーカイブ。これは、特定のバージョン以降に変更されたアーカイブをすべてリストアップする場合に便利です。

また、以下のいずれか 2 つを選択して、レポートのタイプをさらに制限することができます。

- 日付
- リビジョン作成者

- リビジョンをロックしたユーザ
- アーカイブの所有者

履歴レポートを生成できるかどうか、およびどの情報をレポートに使用できるかは、2つのベース権限により規定されます。

- ViewArchiveHeader 権限。この権限はアーカイブ情報を含み、ただしリビジョン情報を含まない履歴レポートの生成に必要です。
- ViewArchiveRev 権限。この権限はリビジョン情報を含む履歴レポートの生成に必要です。

履歴レポートの読み方

デスクトップクライアントで生成され、テキストフォーマットで表示される履歴レポートの例を次に示します。この例は、アーカイブとリビジョンの両方の情報を含む、完全な履歴レポートです。点線の上はアーカイブに関する情報、点線の下はリビジョンに関する情報です。



注 アーカイブに NoGenerateDelta が設定されている場合は、「削除行 / 追加行 / 移動行の数 :」の値が常に、0/0/0 となります。

変更履歴

```

アーカイブ:          C:\Program Files\Serena\vm\common\SampleDB\archives\bridge\bridge.clw-arc
作業ファイル:      bridge.clw
作成されたアーカイブ: 1998/05/18 15:37:40
分割モード:        Unsplit
所有者:            Admin
最新のトランクリビジョン: 1.0
ロック:
グループ:          Development : 1.0
リビジョンの数:    1
属性:
WRITEPROTECT
CHECKLOCK
NOEXCLUSIVELock
NOEXPANDKEYWORDS
NOTRANSLATE
NOCOMPRESSDELTA
NOCOMPRESSWORKIMAGE
GENERATEDELTA
COMMENTPREFIX = " "
NEWLINE = "#r#n"
バージョンラベル:
コメント:
Sample Project Database - first revision.
-----
Rev 1.0
チェックイン日時: 1998/05/18 15:37:40
最終変更日: 1998/05/18 15:37:40
作成者 ID: Admin   削除/追加/移動された行: 0/0/0
Initial revision.
=====

```

↑
アーカイブに関する情報

↓
リビジョンに関する情報

デスクトップクライアントを使用する場合

デスクトップクライアントでは、プロジェクトデータベース、複数または 1 つのプロジェクト、複数のバージョン管理ファイル、または 1 つのバージョン管理ファイルについて履歴レポートを生成できます。プロジェクトデータベースまたはプロジェクトを選択した場合は、次のいずれかについてレポートを生成できます。

- プロジェクトデータベースまたはプロジェクト中の、すべてのプロジェクト / サブプロジェクトのすべてのバージョン管理ファイル
- プロジェクト / サブプロジェクトを含まず、選択されたプロジェクトデータベースまたはプロジェクト中のバージョン管理ファイルのみ



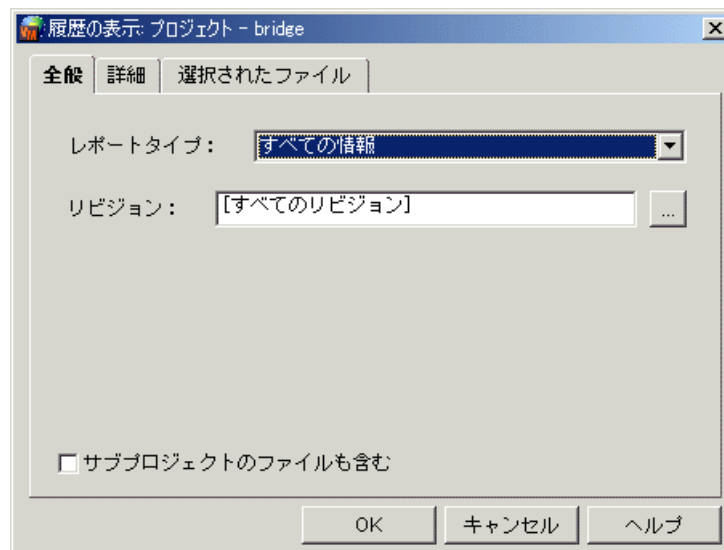
注 5.3 プロジェクトについては履歴レポートは生成できません。プロジェクトの内容についてはレポートを生成できますが、プロジェクト自体については生成できません。

履歴レポートを生成するには

- 1 プロジェクトデータベース、プロジェクト、またはバージョン管理ファイルを選択します。

連続する複数のプロジェクトを選択して履歴レポートを作成するには、最初のプロジェクトを選択して SHIFT キーを押し、そのまま最後のプロジェクトを選択します。連続しない複数のプロジェクトを選択して履歴レポートを作成するには、CTRL キーを押したまま、それらのプロジェクトを 1 つずつクリックします。

- 2 [アクション] - [履歴の表示] を選択します。[履歴の表示] ダイアログボックスが開き、[全般] タブがアクティブになっています。



- 3 【OK】をクリックすると、(アーカイブとリビジョンの両方の情報を含む) 完全なレポートが生成されます。あるいは、[全般] タブ上で次の作業を行ってレポートの内容を制限します。

レポート内容	操作
すべてのリビジョンでなく、特定の リビジョン	〈リビジョン〉 テキストボックスの隣の【...】をクリックし、[リビジョンの選択] ダイアログボックスを表示します。情報を必要とするリビジョンのリビジョン番号、バージョンラベル、またはプロモーショングループを選択します。
アーカイブ情報を表示し、リビジョン 情報は表示しない	〈レポートタイプ〉 ドロップダウンリストから「ファイル情報のみ」を選択します。
アーカイブ情報を表示し、リビジョン 情報は表示しない	〈レポートタイプ〉 ドロップダウンリストから「リビジョン情報のみ」を選択します。
現在ロックされているリビジョン	〈レポートタイプ〉 ドロップダウンリストから「ロックされたリビジョンの情報」を選択します。
指定したバージョンラベルに対応する リビジョン番号	〈レポートタイプ〉 ドロップダウンリストから「バージョンラベルを持つリビジョンの情報」を選択します。次に〈ラベル〉 テキストボックスの隣にある【...】をクリックし、バージョンラベルを選択します。
指定したプロモーショングループに 関連付けられたリビジョン	〈レポートタイプ〉 ドロップダウンリストから「グループのリビジョンの情報」を選択します。次に〈グループ〉 テキストボックスの隣にある【...】をクリックし、プロモーショングループを選択します。
トランク中の最新のリビジョン	〈レポートタイプ〉 ドロップダウンリストから、「最新リビジョンの情報」を選択します。
アーカイブのチップリビジョンが、 指定したリビジョン番号またはバージョンラベルと異なるもの	〈レポートタイプ〉 ドロップダウンリストから、「指定リビジョンではないチップの確認」を選択します。

- 4 日付、作成者、所有者、リビジョンをロックしたユーザ、またはそれらの組み合わせによりさらにレポートを制限する場合は [詳細] タブをクリックします。そうでない場合は手順 6 に進みます。

履歴の表示: プロジェクト - bridge

全般 詳細 選択されたファイル

レポート作成条件

作成者: ...

ロック: ...

所有者: ...

日付の範囲: 開始: ...

終了: ...

OK キャンセル ヘルプ

- 5 [詳細] タブで次のいずれかの作業を行います。

レポート内容	操作
特定のユーザによって作成されたリビジョン	〈作成者〉テキストボックスに作成者のユーザ ID を入力するか、[...] をクリックしてユーザ ID を選択します。複数の ID の間はコンマで区切ります。
特定のユーザにロックされたリビジョン	〈ロック〉テキストボックスにロックしたユーザのユーザ ID を入力するか、[...] をクリックしてユーザ ID を選択します。複数の ID の間はコンマで区切ります。
特定のユーザが所有するアーカイブ	〈所有者〉テキストボックスに所有者のユーザ ID を入力するか、[...] をクリックして所有者を選択します。複数の ID の間はコンマで区切ります。
特定の日付の範囲内においてチェックインされたリビジョン	<p>〈開始日付〉テキストボックスに開始日、〈終了日付〉テキストボックスに終了日を入力します。日付と時刻は、オペレーティングシステムに対して設定したフォーマットで入力する必要があります。</p> <p>Windows では、[コントロールパネル] で [地域] を選択してフォーマットを設定します。</p> <p>UNIX では、PVCS_DATE_FORMAT および PVCS_TIME_FORMAT 環境変数を設定してフォーマットを定義します。</p> <p>コントロールパネルでの設定を行わない場合、あるいは環境変数が設定されていない場合は、mm/dd/yy hh:mm というフォーマットになります。</p>
特定の日付の範囲内においてチェックインされたリビジョン (続き)	<p>注 UNIX 上では、PVCS_DATE_FORMAT を MM/dd/yyyy (日を先にする場合は dd/MM/yyyy) に設定することをお勧めします。ここで、d と y はそれぞれ、日と年を表す数字です。M は月を表す英語の略号です。年は 4 桁で、月は大文字で指定してください。このフォーマットを使用すると、最適な結果が得られます。</p>

- 6 履歴レポートを生成する前に、選択されたバージョン管理ファイルを検証するには [選択されたファイル] タブをクリックします。
- 7 [OK] をクリックします。ジャーナルレポートが生成されて表示されます。

コマンドラインインターフェイスを使用する場合

コマンドラインインターフェイスを使用して履歴レポートを生成するには、VLOG コマンドを使用します。このコマンドの構文は次のとおりです。

```
vlog [option...] [file_name...]
```

ここで、*file_name* は 1 つまたはそれ以上のアーカイブまたは作業ファイルを指定します。


```

USER kasiaj/ (SUPERUSER)
USER Admin/ (SUPERUSER)
USER chrisp/ (UNLIMITED)
USER georges/ (UNLIMITED)
USER ken/$$ken (UNLIMITED)
USER edru/ (UNLIMITED)
USER betsyf/ (UNLIMITED)
USER krisb/ (UNLIMITED)
USER briang/ (UNLIMITED)
USER bruceb/ (UNLIMITED)
USER kerstinb/ (UNLIMITED)
USER brads/ (UNLIMITED)
GROUP Dev (Developers): chrisp edru briang ¥
                        betsyf kasiaj shawnl kerstinb

```

セキュリティレポートの内容は次のとおりです。

- 行の継続には円記号 (¥) が使用されます。
- 権限セット、ユーザ、およびグループはそれぞれ、新しい行に記載され、1 行ごとに 1 項目のみが記載されます。
- 権限セットを定義する行にはそれぞれ、権限セットの名前に続いてその権限セットを定義するベース権限が記載されます。次に例を示します。

```

PRIVILEGE Developers: ¥
LOCKTIP ¥
LOCKNONTIP ¥
.
.
.

```

この例では、権限セットの名前は Developers になります。

- ユーザを定義する行にはそれぞれ、ユーザ ID/ パスワードと、かっこ内に権限または権限セットが含まれます。パスワードは、指定されていない場合もあります。次に例を示します。

```

USER kasiaj/ (SUPERUSER)
USER ken/$$ken (UNLIMITED)

```

ユーザ kasiaj にはパスワードは指定されていません。ユーザ ken にはパスワード \$\$ken が指定されています。



注 パスワードを表示するには Superuser 権限が必要です。

- アクセスリストグループを定義する行にはそれぞれ、グループの名前、かっこ内に権限または権限セット、およびそのグループに割り当てられたユーザが記載されます。次に例を示します。

```

GROUP Dev (Developers): chrisp edru briang ¥
                        betsyf kasiaj shawnl kerstinb

```

この例では、グループの名前は Dev で、Developers の権限が割り当てられています。

- コメント行は、シャープ (#) または感嘆符 (!) が行頭に付いています。

デスクトップクライアントを使用する場合

セキュリティレポートを生成するには、「アクセスコントロールデータベースの表示」(ViewAccessDB) 権限が必要です。

セキュリティレポートを生成するには

- 1 表示するアクセスコントロールデータベースに関連付けられた、プロジェクトデータベースまたはプロジェクトを選択します。
- 2 [管理] - [セキュリティ] - [レポートの表示] を選択します。セキュリティレポートが生成され、表示されます。

コマンドラインインターフェイスを使用する場合

セキュリティレポートを生成するには、「アクセスコントロールデータベースの表示」(ViewAccessDB) 権限が必要です。

コマンドラインインターフェイスを使用してセキュリティレポートを生成するには、READDB コマンドを使用します。



注 readdb は、bin ディレクトリのサブディレクトリ、admin に置かれています。

このコマンドの構文は次のとおりです。

```
readdb [option...]
```

オプション

- a このコマンドにもっともよく使用されるオプションは -a です。

```
readdb -adatabase_name
```

-a オプションは、読み込むアクセスコントロールデータベースの名前を指定します。このオプションを使用しない場合は、VCONFIG -a コマンドによって Version Manager プログラムに埋め込まれた値、または AccessDB ディレクティブによってコンフィグレーションファイル中で指定された名前が使用されます。

- p 現在のユーザのパスワードを表示するには、-p オプションを指定します。SuperUser 権限があると、データベース上のすべてのパスワードが表示されます。

READDB コマンドの詳細については、『Serena ChangeMan Version Manager コマンドラインリファレンスガイド』を参照してください。

プロジェクトに対する変更の表示 (change.log)

サブプロジェクトやバージョン管理ファイルの追加や削除など、プロジェクトの内容に対する変更は、プロジェクトデータベースのルートディレクトリにある change.log ファイルに記録されます。このテキストベースのファイルには、タブで区切られて以下のデータが記録されています。

- 変更の日付

- 変更の時刻
- 変更が加えられた場所のタイムゾーン (プロジェクトの場所に基づく)
- 変更者のユーザ ID
- 変更の性質
- 影響を受けたオブジェクトのエンティティパス (引用符で囲む)
- オブジェクトがバージョン管理ファイルである場合のアーカイブパス (引用符で囲む)

change.log ファイルを表示するには、テキストエディタでファイルを開きます。

第 11 章

Serena Collage との統合

Version Manager の Serena Collage との統合について	308
要件	308
統合機能の設定の概要	309
Collage へのファイルの転送	313
転送したファイルの追跡	315

Version Manager の Serena Collage との統合について

Serena ChangeMan Version Manager と Collage を統合すると、Web アプリケーションの開発チームは、ファイルの管理と導入をすべて自動化することができます。Collage は、Web サーバへの資産の導入をスケジュールするための強力なツールを備えており、Web サイトのテストと公開のサイクルを完全に自動化することを可能にします。Collage の導入管理機能を Version Manager と組み合わせることにより、Web アプリケーションを手動で抽出して、それを Web サーバに転送する必要がなくなります。これらの作業は、すべて Version Manager と Collage によって自動的に実行されます。

統合機能の用途

Version Manager と Collage の統合機能を使用すると、以下を行うことができます。

- Web アプリケーションファイルの Collage への転送を自動化し、Collage にそれらのファイルを Web サーバに導入させる
- 転送するファイルを決定了ら、定期的な転送をスケジュールして、Web アプリケーションファイルを自動的に更新する

統合機能の仕組み

段階	内容
1	ビルドエンジニアまたはプロジェクトリーダーが統合機能を設定します。
2	ファイルを導入する準備が整ったら、ビルドエンジニアまたはプロジェクトリーダーによって、Version Manager から Collage へのファイルの転送が開始されます。
3	セットアップファイルで指定されたプロモーショングループ / ラベルとファイルタイプに一致するファイルは、Collage に転送されます。
4	Collage で、ファイルが導入先フォルダにコピーされます。資産プロパティに転送の詳細が表示されます。 Version Manager で、転送を示すラベルがファイルに付けられます。 転送対象のファイルを追跡するためのログファイルも作成されます。

要件

Collage との統合を有効にするには、以下が必要です。

- Version Manager 6.8.1 以降
- Collage 3.0 以降
- Collage に転送するファイルを含む、Version Manager のプロジェクトデータベース
- Version Manager のファイルの転送先とする Collage プロジェクト

統合機能の設定の概要

Version Manager と Collage をインストールし、転送する Version Manager ファイルを特定したら、統合機能を設定することができます。統合機能は、Version Manager のホストシステムで設定する必要があります。

Version Manager ファイルと Collage プロジェクトとの間の統合機能を設定するには、以下を実行する必要があります。

- Version Manager と Collage の設定
- セットアップファイルの作成
- Collge と Version Manager のログイン情報の設定

Version Manager と Collage の設定

Version Manager を設定するには

- 1 Collage に転送するファイルのグループを特定します。
- 2 次のいずれかの作業を行います。
 - バージョンラベルをファイルに割り当てます。
 - プロモーショングループをファイルに割り当てるか、ファイルを同じプロモーショングループにプロモートします。

バージョンラベルとプロモーショングループの詳細については、『Serena ChangeMan Version Manager ユーザガイド』を参照してください。

Collage を設定するには

- 1 複数の Collage サーバを実行している場合は、どのサーバを使用してファイルを転送するかを決定します。このサーバのホスト名または IP アドレス、また場合によってはポート番号が必要となります。
- 2 ファイルの転送先とする Collage プロジェクトをまだ作成していない場合は、それを行います。
- 3 オプションで、ファイルの転送先とする導入フォルダを Collage で作成することもできます。導入フォルダを事前に作成しない場合は、ファイルの転送時に自動的に作成されます。
- 4 Version Manager から転送するすべてのファイルタイプに対応する MIME タイプと資産タイプが Collage にあることを確認します。ファイルタイプのうち欠如しているものがある場合は、その MIME タイプと資産タイプを作成します。

MIME タイプと資産タイプの表示と作成については、『Serena Collage Project Manager's Guide』を参照してください。

- 5 以下のプロジェクトメタデータフィールドを作成し、それぞれを「文字列」タイプフィールドとして定義します。各フィールドのデフォルトの「バージョン処理」オプションをそのまま使用することもできます。



注 メタデータフィールドは、スペースや大文字の使用も含め、ここに記載されたとおりに作成する必要があります。

メタデータフィールド	内容
VM Project Database (VM プロジェクトデータベース)	ソースの Version Manager プロジェクトデータベースの名前
VM Project Name (VM プロジェクト名)	ソースの Version Manager プロジェクトの名前
VM Revision (VM リビジョン)	Version Manager のリビジョン番号。Collage にも独自のリビジョン番号が維持されます。
VM Check in Description (VM チェックインコメント)	転送されたリビジョンに関連付けられたリビジョンのコメント



重要！ これらのメタデータフィールドを作成する前に統合機能を実行すると、フィールドは自動的に作成されます。ただし、フィールドと正しい資産タイプとの関連付けは自動的に行われず、値の書き込みも行われません。

メタデータフィールドの作成については、『Serena Collage Project Manager's Guide』を参照してください。

- 新しい各メタデータフィールドを Version Manager から転送するファイルタイプに対応する資産タイプに関連付けます。たとえば、**.BMP** ファイルを転送する場合は、メタデータフィールドを Collage の「BMP イメージ」資産タイプと関連付けます。ここでは、メタデータフィールドを [プロパティ] ダイアログボックスのどこに表示するか、つまり [プロパティ] - [全般] ビューまたは [プロパティ] - [メタデータ] ビューのどちらに表示するかを選択することもできます。



注 これらのメタデータフィールドの値がユーザによって変更されないようにするには、〈ユーザによる編集〉オプションで「いいえ」を選択します。こうすることにより、統合機能の実行時以外はフィールドの値が変更されないようになります。

詳細については、『Serena Collage Project Manager's Guide』を参照してください。

セットアップファイルの作成

セットアップファイルは、単純な XML 形式のファイルで、統合を完了するために必要な情報はここで指定します。セットアップファイルのフォーマットは、以下のようになります。

```
<?xml version="1.0" encoding="UTF-8"?>
<VersionManagerXML>
  <ProjectInfo VMPDBRoot="VM Project Database">
    <VMProject>VM Subproject</VMProject>
    <VMRecursive>true or false</VMRecursive>
    <VMRevision>Promotion Group or Label</VMRevision>
    <VMFileType>
      <type>*.file extension</type>
    </VMFileType>
    <VMAppliedLabel>Label after transfer</VMAppliedLabel>
    <CollageProject>Collage Project Name</CollageProject>
    <CollageHost>Collage Host Name or IP:port number
  </CollageHost>
    <CollageLocation>Collage Folder</CollageLocation>
    <Logfile>Log File</Logfile>
  </ProjectInfo>
</VersionManagerXML>
```

Version Manager には、複数のセットアップテンプレートファイルが付属しています。そのうちの 1 つをテキストエディタまたは XML エディタで開いて、セットアップファイルの要素を定義します。セットアップファイルは、必要な数だけ作成することができます。

セットアップファイルを作成するには

- 1 テンプレートファイルのうちの 1 つをテキストエディタまたは XML エディタで開きます。
 - UNIX では、ファイルは `VM_Install_dir/vm/cm/samples` にあります。
 - Windows では、ファイルは `VM_Install_dir\vm\cm\Samples` にあります。
- 2 セットアップファイルの要素を次の表の説明に従って定義します。

要素	説明
VMPDBRoot	必ず設定 Version Manager のプロジェクトデータベース。
VMProject	設定は任意 Version Manager サブプロジェクトのパスと名前。指定しない場合、転送はプロジェクトデータベースのルートから開始されます。次にその例を示します。 /bridge/hlp
VMRecursive	必ず設定 サブプロジェクトのファイルを転送するかどうかを指定します。値は true または false です。
VMRevision	必ず設定 このバージョンラベルまたはプロモーショングループを持つファイルを転送するかどうかを指定します。バージョンラベルとプロモーショングループ名では、大文字と小文字が区別されます。
VMFileType	必ず設定 転送する Version Manager ファイルのタイプ。複数のタイプを指定できます。すべてのファイルタイプを指定するには、*.* と入力します。
VMAppliedLabel	設定は任意 転送後に Version Manager のファイルに適用するラベル。指定しない場合、デフォルトの Collage<Collage Folder><Collage Revision> が使用されます。
CollageProject	必ず設定 転送先の Collage プロジェクトの名前。
CollageHost	必ず設定 転送先の Collage マシン名。これには、ホスト名または IP アドレスおよび（必要な場合は）ポート番号を指定します。例： CMServer:81

要素	説明
CollageLocation	必ず設定 ファイルの転送先とする Collage 導入フォルダ。パスも含めます。このフォルダが存在しない場合は、転送時に自動的に作成されます。 このパスをルートから開始するには、/ を使用します。
LogFile	設定は任意 Version Manager からどのファイルが転送され、どのプロジェクトから転送されたかを表示するためのファイル。[表示] - [フィルタ] - [バージョンラベル] を選択し、セットアップファイルで指定されたラベルを入力すると、どのファイルが転送されたかを確認できます。



注 セットアップテンプレートファイルには、CollageLogin、CollagePassword、VMLogin、および VMPassWord という要素が含まれる場合があります。セットアップモードで実行してこれらの要素を正しく定義するには、要素を削除して、[312 ページの『ログイン情報の設定』](#)の手順に従ってください。

- 3 セットアップファイルで複数の転送を定義するには
 - a <ProjectInfo VMPDBRoot> から </ProjectInfo> までのセクションをコピーします。
 - b コピーしたセクションを、</ProjectInfo> の後の最初の転送の下に貼り付けます。
 - c 前出の表を参照して、2 番目の転送の要素を定義します。
 - d これらの手順を繰り返して、必要な数の転送を作成します。
- 4 ファイルを保存します。

ログイン情報の設定

セットアップファイルを作成したら、転送コマンドをセットアップモードで実行して、Version Manager のプロジェクトデータベースと Collage のプロジェクトに対するログイン情報を取得する必要があります。セットアップファイルには、ユーザ ID と暗号化パスワードの両方のセットが保存されます。

以下の場合には、転送コマンドをセットアップモードで再実行する必要があります。

- どの Version Manager プロジェクトデータベースまたは Collage プロジェクトを指定するかをセットアップファイルで変更する場合
- 追加の転送をセットアップファイルで定義する場合

ログイン情報を設定するには

- 1 コマンドラインで次のコマンドを実行します。


```
runvmcm -s setup.xml
```

 ここで、*setup.xml* はセットアップファイルのパスと名前です。
- 2 プロンプトが表示されたら、Collage プロジェクトのユーザ ID とパスワードを入力します。

- 3 プロンプトが表示されたら、Version Manager プロジェクトデータベースのユーザ ID とパスワードを入力します。



注 Version Manager のログインソースがホスト ID に設定されている場合は、Version Manager のユーザ ID とパスワードの入力を求めるプロンプトは表示されません。

- 4 セットアップファイルで指定した各転送に対して、手順 2 と 3 を繰り返します。

Collage へのファイルの転送

セットアップファイルを作成し、それをセットアップモードで実行したら、次の 3 通りの方法で、Collage へのファイルの転送を開始することができます。

- **手動**: コマンドラインで `runvmcm` コマンドを実行します。
- **自動**: OS でスケジュールされたタスクまたは Collage でスケジュールされたサーバコマンドとして、転送をスケジュールすることができます。
- **Version Manager のデスクトップクライアントを使用**: Version Manager で、転送を開始するためのツールバーボタンを作成します。

手動によるファイルの転送

次のいずれかの作業を行います。

- コマンドラインで次のコマンドを実行します。

```
runvmcm setup.xml
```

ここで、`setup.xml` はセットアップファイルのパスと名前です。

- 複数のセットアップファイルで定義された転送を開始するには、次のコマンドを実行します。

```
runvmcm setup1.xml setup2.xml setup3.xml
```

ここで、`setup1.xml`、`setup2.xml`、および `setup3.xml` は、セットアップファイルのパスと名前です。

ファイルの自動転送

転送は、OS でスケジュールされたタスクまたは Collage でスケジュールされたサーバコマンドとして設定することができます。コマンドラインに関する説明で示したパラメータと同じものを使用します。スケジュールされたタスクまたはサーバコマンドの設定については、OS または Collage のマニュアルを参照してください。

Windows サービスとしての Collage の実行

Collage を Windows サービスとして実行している場合は、次の重要な点を考慮してください。

- Windows サービスの実行に使用されるユーザアカウントが Version Manager のプロジェクトデータベースファイルに対する十分なアクセス権を持っていることを確認します。たとえば、プロジェクトデータベースが当初 UNC ネットワークパスを使用して作成された場合、その

Collage サービスを実行するユーザアカウントはネットワーク Windows のドメインアカウントである必要があります。ローカルシステムアカウントは、UNC パスを解決できません。

- マップされたドライブ文字を使用して作成されたプロジェクトデータベースからファイルを転送することはできません。Collage をサービスとして実行している場合は、Version Manager のプロジェクトデータベースは、当初 UNC または別のネットワークプロトコルを使用して作成されていなければなりません。

Version Manager デスクトップクライアントを使用したファイルの転送

Version Manager では、転送コマンドとセットアップファイルを指定することにより、ファイルの転送を開始するツールバーボタンを作成することができます。

Version Manager を使用して転送を設定するには

- 1 Version Manager で、[管理] - [ツールの設定] を選択し、【追加】ボタンをクリックします。
- 2 転送の名前を〈名前〉フィールドに入力します。
- 3 〈コマンド〉フィールドに、runvmcm と入力します。以下に例を示します。

コマンド : runvmcm

引数 : "D:¥Program Files¥Serena¥vm¥CMSamples¥vmcmsetup.xml"

- 4 〈引数〉フィールドに、セットアップファイルのパスと名前を入力します。【OK】をクリックします。
- 5 転送を開始するには、ツールバーボタンをクリックします。

転送したファイルの追跡

Version Manager から Collage にファイルを転送したら、転送に関するファイルプロパティをプログラムまたはログファイルから表示することができます。

プログラムまたはファイル	転送情報を表示する手順	表示されるプロパティ
Version Manager	<ol style="list-style-type: none"> 1 [表示] - [フィルタ] - [バージョンラベル] を選択します。 2 適用したラベル名 (セットアップファイルで <VMAppliedLabel> として定義された) を入力します。 デフォルトのラベルは、Collage<Collage Folder><Collage Revision> です。 	<p>デフォルトラベルの場合、以下が表示されます。</p> <ul style="list-style-type: none"> ■ ファイルがコピーされた Collage 内の場所 ■ Collage 資産のバージョン
Collage	<ol style="list-style-type: none"> 1 [コンテンツ] - [フォルダの導入] ビューを選択します。 2 ファイルに関する [プロパティ] ダイアログボックスを表示します。 3 Version Manager の情報を表示するメタデータフィールドをどのように設定したかによって、プロパティが次のどちらかのビューで表示されます。 <ul style="list-style-type: none"> • [プロパティ] - [全般] • [プロパティ] - [メタデータ] 	<ul style="list-style-type: none"> ■ ファイルの転送元である Version Manager プロジェクトの名前 ■ ファイルの転送元である Version Manager プロジェクトデータベースの名前 ■ 転送された Version Manager リビジョンの番号 ■ 転送されたリビジョンに関連付けられたリビジョンの説明
ログファイル	<p>テキストエディタでログファイルを開きます。ログファイルの名前と場所は、セットアップファイルで <LogFile> として定義されています。</p> <p>注 コマンド <code>runvmcm -r setup.xml</code> を実行すると、ログファイルを再設定することができます。</p>	<ul style="list-style-type: none"> ■ 転送の日付と時刻 ■ 転送元の Version Manager ファイルとリビジョンの ID ■ 各ファイルがコピーされた Collage 内の場所 ■ Collage 資産のバージョン

第 12 章

Version Manager の SourceSafe 用変換ユーティリティの使用

はじめに	318
制約	319
変換後のアーカイブの場所	319
変換ユーティリティの使用前の準備	320
IDE プロジェクトの変換	327
変換ユーティリティの実行	330
国際化文字列	333
ログファイル	334

はじめに

Serena ChangeMan Version Manager と Microsoft Visual SourceSafe では異なる格納形式が使用されるため、SourceSafe ファイルを Version Manager で使用するには、まずファイルを Version Manager アーカイブに変換する必要があります。Serena では、これを行うための変換ユーティリティを提供しています。

この付録では、vss2vm 変換ユーティリティについて紹介し、それを使用して Visual SourceSafe から Version Manager ヘファイルを移動する方法について説明します。



重要！ このユーティリティを使用する前に、変換するすべてのファイルが Visual SourceSafe にチェックインされていることを確認してください。

変換により次の動作が実行されます。

- 変換する各 SourceSafe ファイルに関する履歴とコメントが保持されます。
- 変換中の SourceSafe プロジェクトに対応した Version Manager プロジェクトが作成されます。
- ユーザにより定義されたラベルとコメントが保持されます (318 ページの『[ユーザ定義ラベル](#)』参照)。
- ブランチが変換されます。
- 変換中に見つかった共有ファイルを使用するようプロジェクトが設定されます。



重要！ 『Serena ChangeMan Version Manager インストールガイド』に記載された対応ネットワーク構成以外の構成で Version Manager を使用している場合、変換は正しく実行されない可能性があります。変換を開始する前に、320 ページの『[変換ユーティリティの使用前の準備](#)』を参照してください。

ユーザ定義ラベル

SourceSafe は、Version Manager とは異なる方法でユーザ定義ラベル (Version Manager ではバージョンラベル) を割り当てます。SourceSafe ファイルのチップ (最新) バージョンにラベルを割り当てるよう選択すると、割り当てる各ラベルごとに新しいバージョン (Version Manager ではリビジョン) のファイルが自動的に作成されます。

たとえば、ラベルをファイル (チップ) のバージョン 1 に割り当てるよう選択すると、SourceSafe によってバージョン 2 のファイルが作成され、ラベルがバージョン 2 に割り当てられます。2 つのラベルをバージョン 1 のファイル (チップ) に割り当てるよう選択すると、バージョン 2 と 3 のファイルが作成され、一方のラベルがバージョン 2 に、もう一方のラベルがバージョン 3 に割り当てられます。

それに対して Version Manager では、ラベルがリビジョンに割り当てられるときに新しいリビジョンのファイルは作成されません。たとえば、ラベルをリビジョン 1.1 に割り当てるよう選択すると、Version Manager によってラベルはリビジョン 1.1 に割り当てられますが、新しいリビジョンは作成されません。2 つのラベルをリビジョン 1.1 に割り当てるよう選択すると、新しいリビジョンが各ラベルに対して作成されるということではなく、両方のラベルがリビジョン 1.1 に割り当てられます。

変換ユーティリティでは、SourceSafe ファイルを変換する際に Version Manager のラベル付けスキームが使用されます。そのため、変換ユーティリティによって SourceSafe バージョンのラベル

が検出された場合、Version Manager のリビジョン番号は増えません。たとえば、バージョン 2、3、および 4 に対するラベルがある場合、変換ユーティリティによってすべてのラベルが同じ Version Manager リビジョンに割り当てられます。

SourceSafe を使用すると、各バージョンのファイルに対するラベルを定義する際に、コメントを入力することができます。Version Manager では、ラベルとそのコメントは、次に示すように Version Manager の変更コメントに含まれています。

これは変更コメントです。
Label: ラベルのコメント
Label2: ラベル 2 のコメント

制約

変換ユーティリティには以下の制約があります。

- ワークスペース設定は変換時に保持されません。
- ブランチのラベルは、それと同じものが変換先のトランクにすでに存在する場合、変換時にブランチリビジョンには適用されません。その代わりに、ブランチラベルは、ブランチリビジョンに関する Version Manager の変更コメントに挿入されます。同じラベルがトランクリビジョンにない場合は、ブランチに適用されます。
- Visual SourceSafe ファイルのロックは変換時に保持されません。
- 最後にインストールされたソースコード制御インターフェイス（Visual SourceSafe または Version Manager SCC インターフェイスなど）しか使用できない開発環境（Visual C++ など）が少なくありません。そのため、ユーザが使用するすべての製品を同時に変換できるよう、計画する必要があります。たとえば、Visual C++ の 2 つのプロジェクトに Visual SourceSafe と Source Code Control インターフェイスの両方を使用するユーザがいるとします。このユーザは、Version Manager Source Code Control インターフェイスをインストールした後は、Visual C++ プロジェクトに対応した SourceSafe プロジェクトにはアクセスできなくなります。
- アーカイブは、リビジョンライブラリとメタデータに分割されません。Version Manager ファイルサーバで使用するためにアーカイブを分割する場合、それを変換後の別ステップとして行う必要があります。詳細については、[158 ページの『リビジョンライブラリの作成』](#)を参照してください。

変換後のアーカイブの場所

SourceSafe ファイルが変換されると、その Version Manager アーカイブは、ファイルの変換時に指定した Version Manager プロジェクトデータベース配下のディレクトリに配置されます。たとえば、d:\test\archives では、d:\test が Version Manager プロジェクトデータベースの場所になります。

SourceSafe データベース全体を変換する場合、変換ユーティリティによって、データベースに存在する各 SourceSafe プロジェクト用にサブディレクトリが作成され、変換後のアーカイブが適切な

ディレクトリに配置されます。たとえば、SourceSafe プロジェクトが以下のような構造をしているとします。

```
$¥  
├─src  
├─ss_new  
└─test
```

この場合、変換ユーティリティによって、Version Manager プロジェクトデータベースの場所の配下にある archives というディレクトリ内に、archives¥src、archives¥ss_new、および archives¥test という 3 つのサブディレクトリが作成されます。

また、これと同じ名前の 3 つの Version Manager プロジェクトもプロジェクトデータベースに作成されます。

プロジェクトを変換する場合は、そのプロジェクトの archives というディレクトリの下に単一のサブディレクトリ（例：archives¥project）が作成され、変換後のアーカイブはそのサブディレクトリに配置されます。また、SourceSafe プロジェクトと同じ名前の Version Manager プロジェクトもプロジェクトデータベースに作成されます。

ほかのプロジェクトと共有されるアーカイブを含む単一のプロジェクトを変換する場合、これらのアーカイブは、アーカイブの変換ユーティリティによって検出された場合に共有マークを付けられますが、プロジェクトの変換時には実際には変換されません。

変換されたアーカイブのインポート中に共有アーカイブが検出された場合、それと同じ名前のアーカイブが存在するかどうかを確認するために、共有アーカイブのもとのパスがチェックされます。同じ名前のアーカイブが存在する場合、アーカイブはプロジェクトにインポートされます。



注 このパスは、Shared.log ファイルに出力され、共有ファイルを示すために生成される .txt ファイルにも出力されます。

アーカイブが見つからない場合、共有アーカイブの親が削除されていないかぎり、それを変換するためにアーカイブの変換ユーティリティが呼び出されます。親が削除されている場合は、共有アーカイブが使用されます。

単一のファイルを変換する場合、変換後のアーカイブは archives というディレクトリに配置されます。このファイルのためのディレクトリは、archives ディレクトリ配下には作成されません。



注 SourceSafe ファイルを変換する際、変換ユーティリティの実行元であるディレクトリの下に convArch というディレクトリが作成されます。このディレクトリには、Version Manager のアーカイブが一時的に格納されます。変換が完了すると、このディレクトリからはログファイル以外のすべてのファイルが自動的に削除されます。

変換ユーティリティの使用前の準備

vss2vm 変換ユーティリティを実行する前に、Visual SourceSafe プロジェクトを Version Manager に変換するために必要な労力を評価し、システム設定が変換プロセスと互換性があることを確認する必要があります。以下のセクションでは、これらのタスクについて説明します。

VSS2VM 変換作業にかかる労力の評価

ここに記載するガイドラインに従うと、以下を実行できます。

- Visual SourceSafe プロジェクトを Version Manager に変換するのに必要な労力を予測する
- 変換の難度を高めるリスク要因を特定する



注 Serena は、変換作業にかかる時間については一切保証していません。

変換を完了するのに必要な時間は、以下のような多くの要素によって決まります。

- SourceSafe データの複雑性（リスク評価が低、中、高、最高のどこにあてはまるか）
- VSS データベースのサイズ
- プロジェクトの数
- アーカイブの数
- 変換を実行するハードウェアとネットワーク（該当する場合）の速度
- ツールおよびデータに対するユーザの熟練度

『Microsoft Visual SourceSafe Best Practices』（英語版）に記載されたガイドラインに従うと、Version Manager への変換作業を比較的すばやく簡単に行える可能性が高くなります。詳細については、下記の URL にアクセスしてください。

<http://msdn.microsoft.com/ssafe/technical/articles.asp>

ツールおよびデータに対するユーザの熟練度

Visual SourceSafe、Version Manager、コマンドライン、スクリプト作成、および SourceSafe データ自体に関する経験が豊富な場合、問題が発生する可能性は低くなります。それと同様に、これらの分野のどれか 1 つでもあまり精通していないものがある場合、問題が起こる可能性は当然高くなります。

Visual SourceSafe や Version Manager に対するユーザの熟練度が足りなかったり、既存のプロジェクトに対するなじみが浅かったりすると、問題のあるファイルを手動でどのように処理するかについての決断を行うのが難しくなります。また、コマンドラインやスクリプト作成の経験も、問題のあるファイルを処理するうえで重要なスキルです。変換を担当するユーザにこれらの重要なスキルのうちどちらか一方でも欠けている場合は、コンサルティングサービスの利用を検討する必要があります。

以下は、ユーザの経験が必要となる重要な分野です。

- Version Manager に関するユーザの経験とスキル
- Visual SourceSafe (VSS) に関するユーザの経験とスキル
- DOS/ コマンドラインに関するユーザの経験とスキル
- ユーザのスクリプト作成スキル
- VSS データに関するユーザの精通度

リスク要因

以下のセクションでは、リスク要因について要約します。変換にかかる時間にこれらのリスク要因がどのように影響するかについては、[324 ページの表 12-1：リスク評価ワークシート](#)を参照してください。

低レベルのリスク

- 最小限のブランチーこれは、最高でも 2 レベルのブランチしかないことを意味します。
- ブランチが削除されていない
- ロールバックがない
- 固定ファイルがない
- アーカイブされたバージョンがない
- 共有がない
- プロジェクト名やファイル名に特殊文字（括弧、\$、& など）が使用されていない
- ユーザが Version Manager と Visual SourceSafe に精通している
- 『Microsoft Visual SourceSafe Best Practices』に従っている
- 最新バージョン以外のバージョンも保存している
- Visual SourceSafe のリポジトリが 500MB 未満である

中レベルのリスク

- 中程度のブランチーこれは、最高でも 4 レベルのブランチしかないことを意味します。
- 一部のブランチが削除されている
- 一部の共有が削除されている
- ロールバックがない
- 固定ファイルがない
- ブランチにアーカイブされたバージョンがない
- 最上位プロジェクトで共有がない
- プロジェクト名やファイル名に特殊文字（括弧、\$、& など）が使用されていない
- ユーザが Version Manager と Visual SourceSafe に精通している
- 『Microsoft Visual SourceSafe Best Practices』に従っている
- 最新バージョン以外のバージョンも保存している
- Visual SourceSafe のリポジトリが 500MB ~ 1GB である

高レベルのリスク

- 多数のブランチーこれは、5 レベル以上のブランチがあることを意味します。
- 一部のブランチが削除されている
- ロールバックがない
- ファイルが固定されている

- 最上位プロジェクトで共有がない
- 共有が削除されている
- バージョンがアーカイブされている
- ユーザが変更コメントやコメントにコードの一部を含めている
- プロジェクト名やファイル名に特殊文字（括弧、\$、& など）が使用されている
- ユーザが Version Manager または Visual SourceSafe のどちらか一方にしか精通していない
- 『Microsoft Visual SourceSafe Best Practices』に従っていない
- 最新バージョン以外のバージョンも保存している
- Visual SourceSafe のリポジトリが 1GB ~ 3GB である

最高レベルのリスク

- 多数のブランチ—これは、5 レベル以上のブランチがあることを意味します。
- 1 つまたは複数の VSS アーカイブで複数のブランチが削除されている
- 共有が削除されている
- ロールバックがある
- ファイルが固定されている
- バージョンがアーカイブされている
- ユーザが変更コメントやコメントにコードの一部を含めている
- プロジェクト名やファイル名に特殊文字（括弧、\$、& など）が使用されている
- 最上位プロジェクトで共有がある
- ユーザの Version Manager と Visual SourceSafe に対する経験がほとんどない、または皆無である
- 『Microsoft Visual SourceSafe Best Practices』のガイドラインが無視されている
- 最新バージョンしか保存されていない（下記の注を参照）
- Visual SourceSafe のリポジトリが 3GB 以上である



注 「最新バージョンだけを保存する」機能を使用すると、作業ファイルの以前のコピーが削除されますが、履歴は残ります。この機能を使用したオブジェクトの取得を実行しても、ファイルの最新バージョンしか取得できないので失敗に終わります。この機能が有効になっている場合は、最新バージョンを取得し、Version Manager の [作業ファイルの追加] オプションを使用して新しいアーカイブを作成してください。

リスク評価ワークシート

次のワークシートは、Visual SourceSafe から Version Manager への変換が正しくタイムリーに完了するかを予測し、プロジェクトのリスクレベルを判定するために使用できます。

「VSS リスク要因」という見出しの下には、SourceSafe の各機能が列挙されています。これらの機能を使用すると、Version Manager の変換時に問題が発生する危険性が高くなります。使用される機能の数が多いほど、変換の難度は高くなります。

上記のリスク要因に関するガイドラインに従って、ワークシートに値を記入してください。



注 グレー表示になっているボックスは使用できません。たとえば、ファイルを固定している場合、最初から高レベルの労力であり、固定ファイルの数が多いほど、必要な労力も多くなります。

表 12-1 : リスク評価ワークシート

VSS リスク要因	低	中	高	最高
ブランチ				
共有が削除されている				
共有アーカイブ				
Visual SourceSafe リポジトリのサイズ				
Version Manager に対するユーザの経験（経験が多い場合は「低」、経験が少ない場合は「最高」を選択します）				
ブランチが削除されている				
プロジェクト名またはファイル名で特殊文字が使用されている				
バージョンがアーカイブされている				
ファイルが固定されている				
コードの一部が変更コメントやコメントで使用されている				
最上位プロジェクトに共有がある				
最新バージョンだけが保存されている				
ロールバックがある				

評価ワークシートの解釈

労力のレベルを判定するには、表に記入したエントリを参照してください。「最高」列にエントリがある場合は、それにかかる労力は最高レベルであると見なすことができます。

右側の列のチェックマークが少ないほど、タスクは簡単になります。「高」と「最高」の列に 4 つ以上のチェックマークが付いている場合は、複雑な VSS プロジェクトデータベースが使用されており、変換作業にオンサイトのコンサルティングサービスが必要となる可能性があることを意味します。



注 以下は、変換タスクにかかる期間を概算したものです。変換はそれぞれ異なります。特定の変換作業が「低」と評価された場合でも、下記の概算値よりも時間がかかる可能性があります。

リスク評価	変換時間の推測値
低	1 日以下
中	1 ~ 5 日間
高	1 ~ 3 週間
最高	3 ~ 6 週間

システムのセットアップ値と設定の検証

変換ユーティリティを実行する前に、システムのセットアップ値、Visual SourceSafe の設定、Version Manager の設定に関して推奨される手順を完了する必要があります。

システムセットアップ値の前提条件

- 1 ワークステーションまたはアクセス可能なネットワークサーバに Visual SourceSafe 5 または 6 および Version Manager 6.5 以降がインストールされている必要があります。
- 2 ワークステーションまたはアクセス可能なネットワークサーバにコマンドラインの実行可能ファイル `ss.exe` がインストールされている必要があります。
- 3 Visual SourceSafe と Version Manager の実行可能ファイルが PATH 文に記述されている必要があります。

<Visual SourceSafe のパスは、`ss.exe` ファイルが保存されているディレクトリを指していなければならない、Version Manager のディレクトリは、`pvcsvmt.exe` ファイルが保存されている >Install Directory¥VM¥Win32¥bin ディレクトリになります。

Version Manager でこの手順をテストするには、コマンドラインプロンプトで「vcs 勃」と入力します。Version Manager のバージョンと著作権が表示されます。

Visual SourceSafe でこの手順をテストするには、コマンドラインプロンプトで「ss」と入力します。Visual SourceSafe のバージョンと著作権が表示されます。

- 4 SSDIR 環境変数を、変換するデータベースの SRCSAFE.INI ファイルが保存されているディレクトリを参照するように設定する必要があります。このディレクトリを見つけるには、SourceSafe で [ファイル] - [データベースを開く] を選択します。ディレクトリのパスがデータベース名の右に表示されます。

SSDIR 環境変数を設定するには、コマンドラインプロンプトで以下のように入力します。

```
set SSDIR=¥¥<server_name>¥share¥vss
```

この手順をテストするには、コマンドラインプロンプトで以下のように入力します。

```
ss dir -r
```

Visual SourceSafe データベースの内容が表示されます。これがサイズの大きなデータベースである場合は、`-r` オプションは使用しないでください。

Visual SourceSafe における前提条件

- 1 SourceSafe データベース内のユーザ ID が、変換が開始されるときにログインしているユーザのネットワーク ID と一致する必要があります。このユーザは、SourceSafe データベースに対して FULL 権限を持っていることが必要です。ユーザの表示と追加には、Visual SourceSafe Administrator が使用されます。

この手順をテストするには、コマンドラインプロンプトで以下のように入力します。

```
ss whoami
```

- 2 Visual SourceSafe Administrator で、「ネットワーク名を使用して自動的にユーザをログインさせる」ように Visual SourceSafe を設定します ([ツール] - [オプション] - [全般] を選択して [全般] タブを表示)。
- 3 SourceSafe データを移動した場合は、変換する SourceSafe データベースを参照するように SRCSAFE.INI ファイルの `Data_Path` 変数を設定し、SourceSafe が正しいデータベースにアクセスできるようにします。

- 4 Visual SourceSafe で、ソースデータベースに対してデータベース分析を実行します。エラーが発生した場合は、Visual SourceSafe でエラーを手動で修正するか、データベースの分析および修正の各機能を実行します。
- 5 チェックアウトの更新日を維持するよう SourceSafe を設定し、Version Manager がもとの更新日を持つリビジョンをチェックインするようにします。これを実行しない場合、Version Manager アーカイブの更新日は、実際の更新日ではなく チェックインの日付になります。更新日を維持するには、Visual SourceSafe Explorer で [ツール] - [オプション] - [ローカルファイル] を選択し、[ローカルファイル] タブを表示します。次に、〈ローカルファイルの日付と時刻〉ドロップダウンリストから「変更時刻」を選択します。
- 6 [ツール] - [オプション] を選択して、〈現在のプロジェクトを基に作業フォルダを設定〉と〈作業フォルダを基にプロジェクトを設定〉の各オプションを選択解除します。これらのオプションが選択されていない場合は、何も行う必要はありません。
- 7 変換ユーティリティを実行する前に SourceSafe を終了します。

Version Manager における前提条件

- 1 Version Manager のマスターコンフィグレーションファイルで、必要なコンフィグレーション設定を指定する必要があります。マスターコンフィグレーションファイルでは、プロモーションモデルが定義されていないはいけません。定義されているプロモーションモデルがある場合、変換中はそれを無効にします。
- 2 マスターコンフィグレーションファイルが埋め込まれている場合は、変換の前に埋め込みを解除します。76 ページの『マスターコンフィグレーションファイルの埋め込み』を参照してください。



注 別のアーカイブ拡張子を使用するには、まずアーカイブを Version Manager フォーマットに変換し、変換後のファイルのアーカイブ拡張子を変更します。アーカイブ拡張子を変更するには、Convertarch ユーティリティを変換後のアーカイブに対して実行します。Convertarch ユーティリティは、Serena アンサーラインから入手できます。また、Version Manager プロジェクトのデフォルトのアーカイブ拡張子も変更して、指定したアーカイブ拡張子が今後作成するアーカイブに適用されるようにする必要があります。

- 3 変換時は、キーワードの展開を無効にする必要があります。こうすることにより、変換元の SourceSafe バージョンに含まれるキーワードが間違っ展開されるのを防ぐことができます。変換が完了したら、このオプションを再び有効にすることができます。この設定をオンに戻すには、Version Manager デスクトップクライアントで [管理] - [アーカイブ属性] を選択するか、コマンドラインインターフェイスで次のように入力します。

```
vcs +pk foo.c-arc
```

ここで、*foo.c-arc* は 1 つまたは複数のアーカイブファイルの名前です。99 ページの『キーワード展開オプション』を参照してください。

- 4 変換時は、Version Manager のアクセスコントロールデータベースは無効にすることを強くお勧めします。アクセスコントロールデータベースを無効にする場合、次の手順を実行する必要があります (Source Safe ユーザ ID を Version Manager のアクセスコントロールデータベースに追加する)。詳細については、235 ページの『セキュリティの無効化』を参照してください。
- 5 Version Manager アクセスコントロールデータベースを無効にした場合は、SourceSafe データベースにあるすべてのユーザ ID を Version Manager アクセスコントロールデータベースに追加する必要があります。アクセスコントロールデータベースを無効にしていなくても、この手順を実行することは推奨されますが、必須ではありません。詳細については、202 ページの『アクセスコントロールデータベースについて』を参照してください。

- 6 SourceSafe プロジェクトから作成される Version Manager プロジェクトを配置するための Version Manager プロジェクトデータベースを作成する必要があります。24 ページの『プロジェクトデータベースの作成』を参照してください。
- 7 環境変数 ISL VINI を、ISLV.INI ファイルを含むディレクトリを指すよう定義します。デフォルトでは、これ C:¥WINDOWS または C:¥WINNT またはです。

IDE プロジェクトの変換

Microsoft のソースコード制御インターフェイス (SCC) 標準に準拠する統合開発環境 (IDE) からソース管理下に置かれたファイルを変換する場合は、ここに概要を示す特定の IDE の手順に従う必要があります。

Visual Studio .NET プロジェクトの変換

Visual Studio .NET を通して Visual SourceSafe に追加されたファイルを変換する場合、次の変換前および後手順を実行する必要があります。

Visual Studio .NET プロジェクトの変換前手順

.NET プロジェクト変換の準備

- 1 Visual SourceSafe から Visual Studio .NET ソリューション全体の最新のリリースを取得します。



注 Visual SourceSafe が登録 SCC プロバイダでなければなりません。最後にインストールしたプロバイダが有効です。

SCC プロバイダを Visual SourceSafe に変更するには、レジストリキー Hkey_Local_Machine | Software | SourceCodeControlProvider を編集します。
文字列 ProviderRegKey を Software¥Microsoft¥SourceSafe に設定して IDE を再起動します。

- 2 次の手順で、IDE プロジェクトを Visual SourceSafe から切断します。
 - a [ファイル] - [ソース管理] - [ソース管理の変更] を選択します。
 - b ソリューションとプロジェクトを選択して【アンバインド】をクリックします。



注 プロジェクトをソース管理に追加した方法によっては、一度のアンバインド操作ですべてのプロジェクトを選択できない可能性があります。すべてのプロジェクトがアンバインドされるまで選択とアンバインド手順を繰り返してください。

- c [ファイル] - [すべてを保存] を選択します。
 - d Visual Studio .NET を閉じます。
- 3 Windows エクスプローラを使用して IDE プロジェクトのローカルコピー (手順 1 で得たファイル) がある場所を参照し、ディレクトリから次の拡張子を持つすべてのファイルを削除します。
 - *.SCC

- *.vspssc
- *.vssccc



注 ソリューションに Web プロジェクトが含まれている場合、Web サーバでプロジェクトを探してすべての *.scc および *.vspssc ファイルを削除する必要があります。

- 4 次のいずれかの方法で、SCC プロバイダとして Version Manager を設定します。
 - まだ Version Manager をインストールしていなければインストールします。SCC プロバイダはインストール時に Version Manager に設定されます。
 - Version Manager がすでにインストールされているが SCC レジストリキーが別のプロバイダで上書きされているか編集されている場合、レジストリキー Hkey_Local_Machine | Software | SourceCodeControlProvider を編集します。
文字列 ProviderRegKey を SOFTWARE\Merant\Integrations\SCC に変更して、IDE を再起動します。
- 5 ソリューションを Version Manager ソース管理に追加します ([ファイル] - [ソース管理] - [ソリューションをソース管理に追加])。追加手順の詳細については、『Version Manager IDE クライアント操作ガイド』を参照してください。



重要！ ソリューションが Web プロジェクトを含む場合、ソリューション全体ではなく個々のプロジェクトをソース管理に追加することをお勧めします。

上記手順を完了したら [330 ページの『変換ユーティリティの実行』](#) を参照してください。

Visual Studio .NET プロジェクトの変換後手順

Visual SourceSafe ファイルを Version Manager アーカイブに変換したら、次の手順に従ってプロセスを完成します。

変換した .NET プロジェクトのファイナライズ

- 1 Windows エクスプローラを使用して変換したアーカイブがある場所を参照し、ディレクトリから次の拡張子を持つすべてのファイルを削除します。
 - *.scc-arc
 - *.vspssc-arc
 - *.vssccc-arc
 - *.sln-arc
 - *.project_type-arc
例 : *.csproj-arc or *.vbproj-arc
- 2 Windows エクスプローラを使用してもとのアーカイブの上に変換したアーカイブをコピーします。ディレクトリ構造が異なる場合は、小さい単位で行う必要があります。
- 3 変換したプロジェクトをテストします。これで Visual Studio .NET を使用して以前のリビジョンと履歴にアクセスすることができます。
- 4 変換したソリューションに追加のワークステーションを接続するには、各ワークステーションから [ファイル] - [ソース管理] - [ソース管理で開く] を選択し、必要に応じてダイアログと

確認メッセージに応えます。手順の詳細については、『Version Manager IDE クライアント操作ガイド』を参照してください。



重要！ ソース管理にソリューション全体ではなく個々のプロジェクトを追加した場合、以下を行います。

- 1 新しいソリューションを作成します。
- 2 [ファイル] - [ソース管理] - [ソース管理からプロジェクトを追加] を選択します。
- 3 追加するプロジェクトを選択して必要に応じてダイアログと確認メッセージに答えます。追加手順の詳細については、『Version Manager IDE クライアント操作ガイド』を参照してください。
- 4 各プロジェクトについて、手順 2 と 3 を繰り返します。
- 5 [ファイル] - [すべてを保存] を選択します。

他の SCC ベース IDE プロジェクトの変換

Visual Studio .NET 以外の IDE を通して Visual SourceSafe に追加されたファイルを変換する場合、次の変換前および後手順を実行する必要があります。



重要！ Visual Studio .NET プロジェクトを変換するには、[327 ページ](#)の『Visual Studio .NET プロジェクトの変換』を参照してください。

他の SCC ベース IDE プロジェクトの変換前手順

IDE プロジェクト変換の準備

- 1 Visual SourceSafe から全 IDE プロジェクトの最新のリビジョンを取得します。



注 Visual SourceSafe が依然登録 SCC プロバイダでなければなりません。最後にインストールしたプロバイダが有効です。

SCC プロバイダを Visual SourceSafe に変更するには、レジストリキー
Hkey_Local_Machine | Software | SourceCodeControlProvider
を編集します。
文字列 ProviderRegKey を
Software\Microsoft\SourceSafe に設定して IDE を再起動します。

- 2 IDE を閉じます。
- 3 Windows エクスプローラを使用して IDE プロジェクトのローカルコピー（手順 1 で得たファイル）がある場所を参照し、ディレクトリから拡張子 *.scc を持つすべてのファイルを削除します。

上記手順を完了したら [330 ページ](#)の『変換ユーティリティの実行』を参照してください。

他の SCC ベース IDE プロジェクトの変換後手順

Visual SourceSafe ファイルを Version Manager アーカイブに変換したら、次の手順に従ってプロセスを完成します。



注 IDE 固有のメニューコマンドや手順の詳細については、『Version Manager IDE クライアント操作ガイド』を参照してください。

変換した IDE プロジェクトのファイナライズ

- 1 Windows エクスプローラを使用して IDE プロジェクト用に変換したアーカイブがある場所を参照し、ディレクトリから拡張子 *.scc を持つすべてのファイルを削除します。
- 2 次のいずれかの方法で Version Manager を SCC プロバイダに設定します。
 - まだ Version Manager をインストールしていなければインストールします。SCC プロバイダはインストール時に Version Manager に設定されます。
 - Version Manager がすでにインストールされているが SCC レジストリキーが別のプロバイダで上書きされているか編集されている場合、レジストリキー
Hkey_Local_Machine | Software | SourceCodeControlProvider
を編集します。
文字列 ProviderRegKey を
SOFTWARE¥Merant¥Integrations¥SCC に変更して IDE を再起動します。
- 3 IDE で IDE のプロジェクトを開きます。
- 4 IDE のプロジェクトをソース管理に追加します。アーカイブがすでに存在するので、ワークフローを既存のアーカイブに新規リビジョンとしてチェックインするか尋ねられます。「はい」と答えてそれを実行します。
- 5 変換したプロジェクトをテストします。これで IDE を使用して以前のリビジョンと履歴にアクセスすることができます。
- 6 必要に応じて変換したプロジェクトに追加のワークステーションを接続します。

変換ユーティリティの実行

変換ユーティリティは、対話モードと非対話モードのどちらでも実行できます。変換ユーティリティを実行する際は、-# オプションを使用して、冗長ログファイルを作成します。このログファイルは、変換時に発生したエラーの性質を判定するうえで役に立ちます。-# オプションを使用しない場合、変換時には最小限のログ情報しか記録されません。



重要！ 変換ユーティリティを実行する前に、変換するすべてのファイルがチェックインされており、すべてのユーザが Visual SourceSafe からログオフしていることを確認します。

変換ユーティリティを対話モードで使用するには、コマンドラインプロンプトで次のように入力します。

```
vss2vm
```

変換ユーティリティにより、変換の実行に必要な情報の指定を求めるプロンプトが表示されます。非対話モードは、変換ユーティリティをスクリプト内で実行するときを使用します。

構文 非対話モードの構文：

```
vss2vm -sr SS_database [/SS_project]  
[-sa SS_file] [-id userid [:pwd]]  
-pr VM_PDB_location [-pp VM_project]  
[-#]
```

対話モードの構文：

```
vss2vm [-id userid] [:pwd] [-#]
```

オプション



重要！ コマンドオプションは小文字で入力してください。

-sr -sr *SS_database* [*/SS_project*]

-sa -sa *SS_archive*

変換対象の SourceSafe データベース、SourceSafe プロジェクト、または単一の SourceSafe ファイルを指定する必要があります。非対話モードを使用する場合、これらのオプションのうち 1 つを指定することが必須です。

SourceSafe データベース全体を変換するには、コマンドラインで -sr\$/ を指定します。

SourceSafe プロジェクトとその配下にあるすべてのプロジェクトを変換するには、コマンドラインで -sr\$/*SS_project* を指定します (例： -sr\$/test)。

単一の SourceSafe ファイルを変換するには、ファイルの所属先であるプロジェクトまたはプロジェクトデータベースとファイルを指定します (例： -sa\$/test/readme)。

-pr -pr *VM_PDB_location*

変換後の SourceSafe プロジェクトとファイルが配置される Version Manager プロジェクトデータベースの場所を指定する必要があります (例： -prd:¥SS_conversion)。これは、非対話モードを使用する場合に必須です。

-pp -pp *VM_project*

このオプションは、変換後の SourceSafe プロジェクトとファイルが配置される、Version Manager プロジェクトデータベース内の Version Manager プロジェクトを指定します。Version Manager プロジェクトを指定する場合、変換する SourceSafe プロジェクトは Version Manager プロジェクトの配下にサブプロジェクトとして配置されます。たとえば、checkers という SourceSafe プロジェクトを変換する場合に、games という Version Manager プロジェクトを指定すると、変換後プロジェクトの checkers は、プロジェクト games のサブプロジェクトになります (/games/checkers)。

-id -id *userid* [*:pwd*]

このオプションの指定が必須となるのは、アクセスコントロールデータベースが有効になっている Version Manager プロジェクトデータベースまたはプロジェクトを指定する場合と、アクセスコントロールデータベースで定義されているユーザ ID にパスワードが関連付けられている場合にユーザ ID やログインソースを取得するために使用されるログインソースが VLOGIN (ログインダイアログ) である場合だけです。これらの場合には、このオプションに値を指定しなければなりません。

- # このオプションは、冗長モードを有効にします。これにより、変換ユーティリティの冗長出力がディスプレイと `ss2pvcs.log` というログファイルに出力されます。デフォルトは、出力を表示しないことです（非冗長モード。このファイルを作成しない）。表示される出力は、ログファイルに書き込まれた情報と同じになります。

変換ユーティリティを対話モードで実行するには



注 変換ユーティリティの実行可能ファイル (`vss2vm`) は移動しないでください。移動すると、変換ユーティリティは実行されません。

- 1 320 ページの『[変換ユーティリティの使用前の準備](#)』に記載された手順を実行します。
- 2 変換ユーティリティによって `convArch` ディレクトリを作成する場所にディレクトリを変更 (`cd`) します (319 ページの『[変換後のアーカイブの場所](#)』参照)。必要に応じて、ディレクトリを作成します。
- 3 次のように入力して、このディレクトリから変換ユーティリティを実行します。

```
vss2vm [-iduserid [:pwd]] [-#]
```

変換ユーティリティによって次のプロンプトが表示されます。
Enter the path to the Version Manager database where the converted archives will be placed:
- 4 Version Manager プロジェクトデータベースのパスを次の例のように入力します。

```
d:¥test
```
- 5 変換ユーティリティにより、2 回目のプロンプトが次のように表示されます。
Do you want to specify a Version Manager project where your archives will be stored? (y/n):
Version Manager プロジェクト名を入力するプロンプトを表示するには「y」を、表示しないようにするには「n」を入力します。
- 6 「y」を入力した場合、変換ユーティリティにより次のプロンプトが表示されます。
Enter the Version Manager project where the converted archives will be placed:
Version Manager プロジェクトの名前を次の例のように入力します。

```
Temp
```

変換ユーティリティによって次のプロンプトが表示されます。
Choose an option:
1.) Convert entire VSS database
2.) Convert one VSS project or subproject
3.) Convert one VSS file
次のいずれかの作業を行います。
 - SourceSafe データベース全体を変換するには「1」を選択します。有効な入力値は「\$/」です。
データベース内のすべてのファイルとプロジェクトが変換されます。変換ユーティリティにより、データベースに存在する各 SourceSafe プロジェクトに対してサブディレクトリが作

成され、変換後のアーカイブはこれらのディレクトリに配置されます（319 ページの『変換後のアーカイブの場所』参照）。

- SourceSafe データベース内の単一のプロジェクトまたはサブプロジェクトを変換するには「2」を選択します。プロジェクトまたはサブプロジェクトの名前を指定する必要があります。有効な入力値は以下のものです。

```
$/project
$/project/subproject
```

変換ユーティリティにより、変換するプロジェクトの名前を入力するプロンプトが表示されます。プロジェクトの名前を入力します。

- SourceSafe データベース内の単一のファイルを変換するには「3」を選択します。ファイルの名前を指定する必要があります。有効な入力値は以下のものです。

```
$/project/filename
$/project/subproject/filename
```

変換ユーティリティにより、ファイルの名前を入力するプロンプトが表示されます。ファイル名を入力します。

変換ユーティリティを非対話モードで実行するには



注 変換ユーティリティの実行可能ファイル（vss2vm）は移動しないでください。移動すると、変換ユーティリティは実行されません。

- 1 320 ページの『変換ユーティリティの使用前の準備』に記載された手順を実行します。
- 2 変換ユーティリティによって convArch ディレクトリを作成する場所にディレクトリを変更（cd）します（319 ページの『変換後のアーカイブの場所』参照）。必要に応じて、ディレクトリを作成します。
- 3 以下のいずれかを行って、このディレクトリから変換ユーティリティを実行します。
 - 非対話モードの vss2vm コマンドラインを入力します（非対話モードの構文については、331 ページを参照）。

次にその例を示します。

```
vss2vm -sr$/test -prd:¥sample
```

この例では、test という名前の SourceSafe プロジェクトが変換され、d:¥sample にある Version Manager データベースに配置されます。

- vss2vm コマンドラインを含むスクリプトを作成します。以下は、conversion.pcli という名前の Version Manager PCLI スクリプトの例です。

```
set -vSSprj $1
set -vVMpdb $2
run -e vss2vm -sr'$/'$SSprj -pr$VMpdb
```

国際化文字列

このユーティリティは、国際化文字列がメッセージカタログ（pvcsvm.cat）に含まれる場合にそれを使用するよう設計されています。変換ユーティリティがカタログファイルを表示できるようにするには、環境変数 NLSPATH をカタログファイルの場所に設定する必要があります。デフォルトで、カ

タログファイルは、Version Manager がインストールされている場所の bin ディレクトリにあります（例：C:\Program Files\Serena\vm\common\bin\win32）。

NLSPATH 変数を設定するには

- 1 [スタート] – [コントロールパネル] – [システム] の順に選択します。[システムのプロパティ] ダイアログボックスが表示されます。
- 2 [詳細設定] タブをクリックします。
- 3 [環境変数] ボタンをクリックします。[環境変数] ダイアログボックスが開きます。
- 4 [新規] をクリックします。[新しいシステム変数] ダイアログボックスが表示されます。
- 5 〈変数名〉フィールドに「NLSPATH」と入力します。
- 6 〈変数値〉フィールドで、pvcsvm.cat ファイルの場所を指定します。
- 7 各ダイアログボックスで【OK】ボタンをクリックします。

文字列がメッセージカタログに含まれることを確認するには、コマンドプロンプトから `vss2vm -#` を実行します。

文字列が英語で表示される場合は、カタログファイルに適切な文字列が含まれていないか、NLSPATH が正しくありません。



注 ローカライズされた Visual SourceSafe プロジェクトを変換する場合は、変換を開始する前に Serena サポートにお問い合わせください。ご使用の言語専用のメッセージカタログファイルを提供させていただきます。

ログファイル

変換ユーティリティは、SourceSafe ファイルの変換時に 4 つのログファイルを作成します。

デフォルトで、これらのファイルは、それぞれ `ss2pvcs.log`、`ConvErr.log`、`shared.log`、および `dgb_path.log` という名前で `convArch` ディレクトリに保存されます。

変換ユーティリティは、その実行元であるディレクトリの下に `convArch` というディレクトリを作成します。

変換が完了すると、`.log` ファイルと `.cfg` ファイルを除き、`convArch` ディレクトリの内容は削除されます。

変換が失敗した場合や、完了する前に中断された場合、`convArch` の内容は削除されません。`convArch` ディレクトリを削除せずに変換を再実行すると、`convArch0` という別のディレクトリが作成され、変換はこのディレクトリで行われます。既存の `convArch` ディレクトリを削除しないと、変換を実行するたびに新しい `convArch` ディレクトリが作成されます（`convArch1`、`convArch2` など）。

ss2pvcs.log

`ss2pvcs.log` ファイルには以下の情報が含まれます。

- 変換された SourceSafe ファイルの名前
- 変換後のファイルが保存されている Version Manager アーカイブの場所

- 変換時に変換ユーティリティによって実行された動作（アーカイブの検索、アーカイブのインポート、プロジェクトの作成など）

convERR.log

convERR.log ファイルは、変換プログラムがアーカイブを Version Manager プロジェクトデータベースにインポートするときに生成されます。通常、このファイルは空白ですが、変換ユーティリティが冗長モードで実行されている場合、いくつかの情報がこのファイルに記録されます。このファイルの情報は、主に情報目的のために提供されるもので、エラーの内容などは含まれません。たとえば、このログファイルには、パスやファイル名が記録されます。

Shared.log

shared.log ファイルには、共有ファイルへの参照と共有元の場所が記録されます。

```
"path¥filename" shared from "parentpath¥filename"
```

共有ファイルを示すためのテキストファイルも生成されます。[319 ページの『変換後のアーカイブの場所』](#)を参照してください。

dgb_path.log

このファイルは、Serena サポートが使用します。

索引

記号

\$HOME 21
\$Log\$ キーワード 100, 112

A

AccessList ディレクティブ 84
Adobe Acrobat 12
AllEvents 272
ArchiveWork ディレクティブ 111
AutoCreate ディレクティブ 84

B

BaseVersion ディレクティブ 265
BranchVersion ディレクティブ 265
Builder、との統合 24

C

change.log ファイル 304
COBOL ソース 113
Collage
 統合 307
 統合機能の設定 309
 ファイルの転送 313
Collage との統合 307
ColumnMask ディレクティブ 113
CommentPrefix ディレクティブ 112
convERR.log 335

D

default.cfg 72, 76
DefaultVersion ディレクティブ 265
DeleteWork ディレクティブ 98
delta.d 132
dgb_path.log 335

E

EOL シーケンス、クロスプラットフォーム使用時の変換
 コマンドラインインターフェイス 66

 デスクトップクライアントの場合 48
EventArchive 277
EventArchiveName 277
EventChgDesc 277
EventClientWorkfile 277
EventConfigFiles 277
EventDate 277
EventEntityPath 277
EventFolderPath 277
EventFQPWorkfile 277
EventGroup 277
EventItemSelectionType 277
EventJournalEntry 277
EventJournalFile 277
EventLock 277
EventMasterConfigFile 277
EventName 277
EventParmFile 277, 284
EventPassword 277
EventProjectDB 278
EventProjectName 278
EventRevision 278
EventTime 278
EventUserID 278
EventVersion 278
EventWorkfile 278
ExclusiveLock ディレクティブ 84
ExpandKeywords Touch ディレクティブ 101
ExpandKeywords ディレクティブ 112

F

File Server 129
 SSL の有効化 167
 VTRANSFER 163
 アーカイブの移動 163
 アーカイブのインポート 163
 アーカイブのエクスポート 163
 アーカイブの修正 163
 アーカイブの名前変更 163
 アーカイブの分割解除 160
 インストール 134
 管理 134
 管理者、管理 136
 管理ユーティリティ
 起動 136
 概要 130
 起動と停止 134

共有アーカイブ 149
クロスプラットフォームに関する特記事項 165
互換性 131
仕組み 132
ステータスの表示 146
セキュリティ 166
デフォルトの管理パスワード 137
デルタ 132
複数の File Server 142
利点 130
リビジョンライブラリ 158
ログ 148
ForceUnlock ディレクティブ 86

H

HOME 環境変数 21
Host Id 90
HTML レポート 292

I

Issue の関連付け 28

J

Journal ディレクティブ 89

L

LDAP
 Microsoft の例 98
 Netscape の例 98
 コンフィグレーション 95
 冗長サーバ 95
 チェックインオプション 96
 ポート番号 96
LDAP Id 90
Lightweight Directory Access Protocol 95
LogIn ディレクティブ 90

M

MAKEDB コマンド 225
Microsoft Active Directory と LDAP 90, 98
Microsoft Windows ネットワーク 91
Mover、接続 125
MultiLock ディレクティブ 86, 268

N

Netware Id 91
NetWare セマフォ 109
NewLine ディレクティブ 112
nfsmap ファイルの編集 43, 63
NoJournal ディレクティブ 89
nonsetuid 20

O

Owner ディレクティブ 84

P

PathSeparator ディレクティブ 101
PostCreateArchive 273
PostGet 273
PostLock 273
PostPromote 273
PostPut 272
PostUnlock 273
PostVersionLabel 273
PreCreateArchive 273
PreGet 272
PreLock 273
PrePromote 273
PrePut 272
PreUnlock 273
PreVersionLabel 273
PVCS_BINDIR 64
PVCS_DATE_FORMAT 215
PVCS_TIME_FORMAT 215

Q

Quiet ディレクティブ 106

R

RecordLength ディレクティブ 112
ReferenceDir ディレクティブ 103
Renummer ディレクティブ 113
TrackerLink の設定 122

S

Secure Socket Layer
 File Server における有効化 167
SemaphoreDelay ディレクティブ 108
SemaphoreDir ディレクティブ 108
SemaphoreRetry ディレクティブ 108

SemSuffix ディレクティブ 109
Serena Builder、との統合 24
Serena ChangeMan Mover、接続 125
Serena、お問い合わせ 13
setuid
 UNIX 20
 無効化 43, 63
Shared.log 320, 335
SignOn ディレクティブ 106
SourceBridge
 Issue の関連付け 28
SourceBridge、セットアップ 122
SourceSafe データベースの変換 23, 317
ss2vm.log 334
SSL
 File Server における有効化 167

T

TeamTrack
 Issue の関連付け 28
TrackerLink
 Issue の関連付け 28

U

UnconditionalPrePut 272
UNIX
 nfsmap ファイル 43, 63
 PVCS_BINDIR の設定 64
 setuid 20
 クロスプラットフォームの場合のセットアップ
 デスクトップクライアントの場合 41
UNIX プラットフォーム
 Version Manager Web クライアントの実行、起
 動、および停止 135

V

VCONFIG コマンド 94, 119
VCS Id 91
VCS コマンド 265
vcs.cfg ファイル 79
VCSCFG 環境変数 79
VCSDir ディレクティブ 104
VCSEdit ディレクティブ 106
VCSID 環境変数 91
Verbose ディレクティブ 106
Version Manager File Server
 VTRANSFER 163
File Server
 アーカイブの分割 158
Version Manager

Collage との統合 307
Mover へのログイン 126
セットアップ
 コマンドラインインターフェイス 53
 デスクトップクライアントの場合 15
プロジェクトデータベースについて 17
プロジェクトの説明 15
マスターコンフィグレーションファイルの埋め込
 み 60
Version Manager Development Interface 22
Version Manager File Server 129
 delta-todo.d 132
 SSL の有効化 167
 VTRANSFER 163
 アーカイブの移動 163
 アーカイブのインポート 163
 アーカイブのエクスポート 163
 アーカイブの修正 163
 アーカイブの名前変更 163
 アーカイブの分割 158
 アーカイブの分割解除 160
 インストール 134
 管理 134
 管理者、管理 136
 管理ユーティリティ
 起動 136
 概要 130
 起動と停止 134
 共有アーカイブ 149
 クロスプラットフォームに関する特記事項 165
 互換性 131
 仕組み 132
 セキュリティ 166
 デフォルトの管理パスワード 137
 デルタ 132
 表示、ステータス 146
 複数の File Server 142
 利点 130
 リビジョンライブラリ 158
 ログ 148
Version Manager Web クライアント
 UNIX での実行、起動、停止 135
Version Manager の SourceSafe 用変換ユーティリ
 ティ 317
VJOURNAL コマンド 296
VLOGIN 90
VLOG コマンド 301
VSPLIT 158

W

Windows
 クロスプラットフォームの場合のセットアップ
 コマンドラインインターフェイス 60
 デスクトップクライアントの場合 41

ネットワーク 91
Windows プラットフォーム
Version Manager Web クライアント 135
WNet ID 91
WorkDir ディレクティブ 111
WriteProtect ディレクティブ 84

あ

アーカイブ
UNIX の場合 20
アイテムの移動 192
アクセスの制限 205
インポート 193
書き込み保護 84
共有 21, 32
クロスプラットフォームでの使用のための書き込み許可 46
検索パスのオプション 104
コマンドラインインターフェイスでの作成 57
作成オプション 84
所有者 84
自動作成 84
テンポラリファイル 110
同時アクセスの防止 108
ロック 86
アーカイブ属性
判断 187
変更 185
アーカイブのインポート 193
アーカイブの書き込み保護 84
アーカイブの共有 21, 32
アーカイブの所有者 84
アーカイブの場所
計画
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 20
変更 192
アーカイブレポート。「履歴レポート」参照
アイコンイベント 274
アイテムの移動
 アーカイブ 192
 コンフィグレーションファイル 79
 ツールのコンフィグレーションファイル 185
アクセス、SourceBridge の設定 122
アクセス、TrackerLink の設定 122
アクセスコントロールテキストファイル
 作成 223
 フォーマット 223
 ルール 223
アクセスコントロールデータベース
 アクセスリストグループの定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 218
埋め込み

 コマンドラインインターフェイス 228
 デスクトップクライアントの場合 227
概要 202
権限セットの定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 212
作成
 コマンドラインインターフェイス 225
 デスクトップクライアントの場合 204
デフォルト 204
場所の指定
 デスクトップクライアントの場合 210
プロジェクトデータベースに関連付けられたものの変更 236
プロジェクトとの関連付け 210
無効化 235
ユーザの削除 230
ユーザの定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 214
有効化
 コマンドラインインターフェイス 226
 デスクトップクライアントの場合 221
アクセスリスト
 概要 205
 定義
 コマンドラインインターフェイス 226
 デスクトップクライアントの場合 220
 定義の変更 235
アクセスリストグループ
 解除 234
 概要 205
 権限の割り当て
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 219
 構文、コマンドラインインターフェイス 224
 定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 218
 名前に使用できない文字 219
 メンバーの変更 233
 メンバーの割り当て
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 219
アップグレード
 5.3/6.0 プロジェクトのコピー 33

い

イベント
 アーカイブの作成後 273
 アーカイブの作成前 273
 アイコンとメニュー項目 274
 概要 272
 処理 272

- チェックアウトの後 273
- チェックアウトの前 272
- チェックインの後 272
- チェックインの前 272
- バージョンラベルの適用後 273
- バージョンラベルの適用前 273
- プロモート の後 273
- プロモート の前 273
- ロック解除の後 273
- ロック解除の前 273
- ロックの後 273
- ロックの前 273
- イベント 情報
 - アイコンおよびメニュー項目のイベント 280
 - 取得可能 279
 - リスト 276
 - 渡される方法 281
 - 渡し方
 - 環境変数の使用 281
 - コマンドラインマクロの使用 282
 - パラメータファイルによる方法 284
- イベントトリガ
 - 「 イベント 情報」と「 イベント」も参照
 - 概要 272
 - 情報の転送 276
 - セットアップ
 - コマンドラインインターフェイス 275
 - デスクトップクライアントの場合 274
 - タイプ 272
 - 例 286
- 印刷マニュアル、購入 13

う

- 埋め込み
 - アクセスコントロールデータベース
 - コマンドラインインターフェイス 228
 - デスクトップクライアントの場合 227
 - マスターコンフィグレーションファイル
 - コマンドラインインターフェイス 119
 - デスクトップクライアントの場合 118
 - ログインソース 93
 - コマンドラインインターフェイス 94
 - デスクトップクライアントの場合 94

お

- 大文字と小文字の区別
 - ユーザ ID
 - コマンドラインインターフェイス 66
 - デスクトップクライアントの場合 50
- オプション。「コンフィグレーションオプション」参照
- オンラインドキュメント 12

- オンラインヘルプ
 - アクセス 11
 - コマンドラインインターフェイス 11
 - デスクトップクライアント 11
- オンラインマニュアル
 - 印刷マニュアルの購入 13

か

- 解除
 - アクセスコントロールデータベースのアクセスリストグループ 234
 - アクセスコントロールデータベースのユーザ 230
 - プロジェクト データベースのアクセスコントロールデータベース 238
- ロック 86
- 拡張子
 - アーカイブ 112
 - セマフォファイル 109
 - メッセージファイル 106
- カスタマイズ
 - HTML レポート のフォーマット 292
 - ツールバー 182
 - メニューバー 182
- カスタム権限セット
 - 構文
 - コマンドラインインターフェイス 224
 - 定義
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 212
 - 名前に使用できない文字 213
- カスタムツール 182, 274
- カラムマスク 113
- 環境変数
 - HOME 21
 - PVCS_BINDIR 64
 - PVCS_DATE_FORMAT 215
 - PVCS_TIME_FORMAT 215
 - VCSCFG 79
 - VCSID 91
- 関連付け
 - アクセスコントロールデータベースとプロジェクト 210
 - コンフィグレーションファイルとプロジェクト データベースまたはプロジェクト 77, 82

き

- キーワード
 - \$Log\$ 100, 112
 - EOL シーケンスのインジケータ 112
 - 概要 99
 - コメント 接頭辞 112
 - 展開 112

パス区切り文字 101
キーワード 展開オプション 99
キーワードのパス区切り文字 101
規則、表記上 11
共有アーカイブの検索 149
禁止権限 213

く

クロスプラットフォーム
PVCS_BINDIR の設定 64
計画
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 23
権限の割り当て
 デスクトップクライアントの場合 44
セットアップ
 コマンドラインインターフェイス 60
 デスクトップクライアントの場合 41
グループ。「アクセスリストグループ」と「プロモーショングループ」参照

け

計画
 Version Manager のセットアップ
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 19
 アーカイブの場所
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 20
 クロスプラットフォームの使用
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 23
 作業ファイルの場所
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 20
 セキュリティ 208
 プロジェクトデータベース 19
権限
 アクセスリストグループに対する変更 232
 アクセスリストグループへの割り当て
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 219
 カスタムセットの定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 212
 概要 206
 禁止 213
 クロスプラットフォームで使用するよう割り当て
 コマンドラインインターフェイス 64
 デスクトップクライアントの場合 44
 コンポジット
 概要 206

セット
 概要 207
 ベースファイル
 概要 206
 ユーザに対する変更 231
 ユーザへの割り当て
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 216
 ルール 207
権限セット
 概要 207
 定義
 コマンドラインインターフェイス 224
 デスクトップクライアントの場合 212
 リスト 243

こ

購入、印刷マニュアル 13
固定長ファイル、レコードの長さ 112
コピー
 5.3/6.0 プロジェクトのコピー 22, 33
 5.3/6.0 プロジェクトルート 37
コマンドラインインターフェイス
 Version Manager のセットアップ計画 56
 クロスプラットフォーム用の計画 56
コマンドラインエディタ、指定 106
コマンドラインオプション 106
コメントファイル
 拡張子 106
 概要 106
コンフィグレーションオプション
 アーカイブ検索パス 104
 アーカイブ作成 84
 概要 70
 キーワード 展開 99
 禁止 116
 コマンドライン 106
 作業ファイル属性 98
 参照用ディレクトリ 102
 ジャーナルファイル 89
 設定
 コマンドラインインターフェイス 83, 264
 デスクトップクライアントの場合 82
 設定のガイドライン
 コマンドラインインターフェイス 80
 デスクトップクライアントの場合 76
 セマフォ 108
 テンポラリファイル 110
 ブランチ 262
 ロック 86
コンフィグレーションファイル
 default.cfg 72, 76
 アイテムの移動 79
 使用される場合 79

条件文 84
ツール 183
名前の変更 78
プロジェクト データベースまたはプロジェクトとの関連付け 77
マスター 75, 80
マスターの埋め込み 60
読み込まれる方法
 コマンドラインインターフェイス 81
 デスクトップクライアントの場合 77
理解
 コマンドラインインターフェイス 79
 デスクトップクライアントの場合 75
ローカル 79
ローカルの作成 58
コンポジット 権限
 概要 206

さ

サーバコンフィグレーションファイル 96
最下位のプロモーショングループの定義 181
作業ファイル
 計画
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 19
 チェックイン
 コマンドラインインターフェイス 58
 メッセージファイルの使用 106
 プロジェクトへの追加 28
作業ファイル属性のオプション 98
作業ファイルの場所
 \$HOME 21
 計画
 コマンドラインインターフェイス 56
 デスクトップクライアントの場合 20

削除

アクセスコントロールデータベースのアクセスリストグループ 234
アクセスコントロールデータベースのユーザ 230
チェックイン後の作業ファイル 98

作成

アーカイブ、自動 84
アーカイブディレクトリ
 コマンドラインインターフェイス 57
アクセスコントロールテキストファイル 223
アクセスコントロールデータベース
 コマンドラインインターフェイス 225
 デスクトップクライアントの場合 204
ジャーナルレポート
 コマンドラインインターフェイス 296
 デスクトップクライアントの場合 294
セキュリティレポート
 コマンドラインインターフェイス 304
 デスクトップクライアントの場合 304

パブリックワークスペース 179
ランチ 260
プロジェクト 26
プロジェクトデータベース 24
履歴レポート
 コマンドラインインターフェイス 301
 デスクトップクライアントの場合 299
ローカルコンフィグレーションファイル 58
参照用ディレクトリ
 概要 102
 チェックインオプション 102
 場所 103

し

指定

アクセスコントロールデータベースの場所
 コマンドラインインターフェイス 225
 デスクトップクライアントの場合 210
コマンドラインエディタ 106
ユーザパスワードの指定 215, 217
レポートのフォーマット 290

修正

アクセスリストグループ権限 232
アクセスリストグループのメンバー 233
アクセスリストの定義 235
ユーザ権限 231

使用

作業ファイルのチェックイン時のメッセージファイル 106
セキュリティ 202
プロモーションモデル 246
レポート 289

使用できない文字

アクセスリストグループの名前 219
カスタム権限セットの名前 213
プロジェクトデータベース名 25
プロジェクト名 27

シングルサインオン 91

時刻のフォーマット 215

自動ランチ

概要 261
セットアップ
 コマンドラインインターフェイス 264
 デスクトップクライアントの場合 262

自動マージ 269

ジャーナルファイル

概要 89
チェックインオプション 89

ジャーナルレポート

概要 293
ジャーナルレポートの読み方 293

生成

コマンドラインインターフェイス 296
デスクトップクライアントの場合 294

条件文 84
冗長な LDAP サーバ 95

せ

制御、リビジョンアクセス 86
制限

- アーカイブへのアクセス 205
- パブリックワークスペースの削除 178
- パブリックワークスペースの作成 178
- パブリックワークスペースの名称変更 178
- プロジェクトデータベースの作成 17, 206, 228
- プロモーショングループによるアクセスと権限 256

生成

- ジャーナルレポート
 - コマンドラインインターフェイス 296
 - デスクトップクライアントの場合 294
- セキュリティレポート
 - コマンドラインインターフェイス 304
 - デスクトップクライアントの場合 304
- 履歴レポート
 - コマンドラインインターフェイス 301
 - デスクトップクライアントの場合 299

製品サポート 13

セキュリティ

- アクセスコントロールデータベース 202
- アクセスリスト 205
- アクセスリストグループ 205
- 概要 202
- 計画 208
- セットアップ
 - コマンドラインインターフェイス 222
 - デスクトップクライアントの場合 209
- プロモーションモデル 256
- 無効化 235
- メンテナンス 229
- 有効化
 - コマンドラインインターフェイス 226
 - デスクトップクライアントの場合 221

例 208

セキュリティ、Version Manager File Server 166

セキュリティレポート

- 概要 302
- ジャーナルレポートの読み方 302
- 生成

- コマンドラインインターフェイス 304
- デスクトップクライアントの場合 304

設定

- コンフィグレーションオプション
 - コマンドラインインターフェイス 83, 101, 264
 - デスクトップクライアントの場合 82
- レポートオプション 290
- ワークスペース 181

セットアップ

Version Manager

- コマンドラインインターフェイス 53
- デスクトップクライアントの場合 15

イベントトリガ

- コマンドラインインターフェイス 275
- デスクトップクライアントの場合 274

クロスプラットフォームの使用

- コマンドラインインターフェイス 60
- デスクトップクライアントの場合 41

自動ブランチ

- コマンドラインインターフェイス 264
- デスクトップクライアントの場合 262

セキュリティ

- コマンドラインインターフェイス 222
- デスクトップクライアントの場合 209

ネットワークを使用した複数ユーザのプロジェクト 57

複数のロック

- コマンドラインインターフェイス 268
- デスクトップクライアントの場合 267

プロジェクトデータベース 24

セマフォ

NetWare 109

拡張子 109

概要 108

作成されるディレクトリ 108

試行回数 108

試行間隔 108

チェックインオプション 108

ち

チェックイン、作業ファイル

- コマンドラインインターフェイス 58

つ

追加

- プロジェクトのディレクトリ 28
- カスタムツール 182, 274
- 作業ファイルへのバージョンラベルの割り当て 31
- プロジェクトの作業ファイル 28
- ツール、カスタムツール 182, 274
- ツールのコンフィグレーションファイル
 - アイテムの移動 185
 - デフォルト 183
- ツールバー、カスタマイズ 182
- ツールメニュー 183

て

定義

- アクセスリスト
 - コマンドラインインターフェイス 226
 - デスクトップクライアントの場合 85
- アクセスリストグループ
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 218
- カスタム権限セット
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 212
- 既存のアーカイブのアクセスリスト
 - コマンドラインインターフェイス 226
 - デスクトップクライアントの場合 220
- プロモーションモデル
 - コマンドラインインターフェイス 251
 - デスクトップクライアントの場合 249
- ユーザ
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 214
- 定義、最下位のプロモーショングループ 181
- 定義、デフォルト プロモーショングループ 174
- 展開、キーワード 112
- テンポラリファイル
 - アーカイブ 110
 - 概要 110
 - 作成場所 111
 - チェックインオプション 110
 - デルタ 処理 110
- ディレクティブ
 - AccessList 84
 - ArchiveWork 111
 - AutoCreate 84
 - BaseVersion 265
 - BranchVersion 265
 - ColumnMask 113
 - CommandPrefix 112
 - DefaultVersion 265
 - DeleteWork 98
 - ExclusiveLock 84
 - ExpandKeywords 112
 - ExpandKeywords Touch 101
 - ForceUnlock 86
 - Journal 89
 - LogIn 90
 - MultiLock 86, 268
 - NewLine 112
 - NoJournal 89
 - Owner 84
 - PathSeparator 101
 - Quiet 106
 - RecordLength 112
 - ReferenceDir 103
 - ReNumber 113
 - SemaphoreDelay 108
 - SemaphoreDir 108
 - SemaphoreRetry 108
 - SemSuffix 109

- SignOn 106
- VCSDir 104
- VCSEdit 106
- Verbose 106
- WorkDir 111
- WriteProtect 84
- directives, LogIn 90
- ディレクトリ
 - デフォルト で作成 17
 - プロジェクト への追加 28
- デフォルト コンフィグレーション設定 72, 74
- デフォルト プロモーショングループ
 - 選択 181
 - 定義 174
- デフォルト ユーザ名 126
- デルタの保存 113, 132

と

- 統合、Serena Builder との 24
- 取り出し、アーカイブファイル 110

な

- 名前の変更、コンフィグレーションファイル 78

は

- バージョンラベル
 - 割り当て
 - 作業ファイルの追加 31
- パスワード
 - デフォルト 126
- パラメータファイル 284

ひ

- 引数、イベントトリガへの転送 276
- 日付のフォーマット 215
- 表記上の規則 11
- 表示、プロジェクトに対する変更 304
- 表示、ワークスペースの設定 176

ふ

- ファイルの導入 307
- 復元、アーカイブファイル 110
- 複数のロック
 - 概要 84
 - セットアップ
 - コマンドラインインターフェイス 268
 - デスクトップクライアントの場合 267

- 並行開発 266
- 複製、ユーザの定義 217
- ブランチ
 - 概要 260
 - 自動 261
 - 自動の設定
 - コマンドラインインターフェイス 264
 - デスクトップクライアントの場合 262
 - 多重ロックの使用 266
 - チェックインオプション 262
 - 例 260
- ブランチ、作成される場合 260
- プロジェクト
 - 5.3/6.0 のコピー 22, 33
 - change.log ファイル 304
 - アクセスコントロールデータベースとの関連付け 210
 - アクセス不可 222
 - 概要 15
 - コピー 195
 - コマンドラインインターフェイスにおける組織 55
 - コンフィグレーションファイルとの関連付け 82
 - 作成 26
 - 名前に使用できない文字 27
- プロジェクトデータベース 198
 - アクセスコントロールデータベースとの関連付けの削除 238
 - 概要 17
 - 計画 19
 - コピー 198
 - コンフィグレーションファイルとの関連付け 82
 - 作成の制限 17, 206, 228
 - セットアップ 24
 - 名前に使用できない文字 25
- プロジェクトデータベース、作成 24
- プロジェクトに対する変更の表示 304
- プロジェクトのコピー 195
- プロジェクトルート、コピー 37
- プロモーショングループ
 - アクセスと権限の制限 256
 - 概要 246
- プロモーションモデル
 - 階層 248
 - 概要 246
 - セキュリティ 256
 - 定義
 - コマンドラインインターフェイス 251
 - デスクトップクライアントの場合 249
 - 並行開発 252
 - ライフサイクルの管理 251
 - ルール 246

- 変換、Microsoft Visual SourceSafe データベース 23
- 変換、SourceSafe データベース 23
- 変換、Visual SourceSafe データベース 23, 317
- 変換、クロスプラットフォーム使用時の EOL シーケンス
 - コマンドラインインターフェイス 66
 - デスクトップクライアントの場合 48
- 変更
 - アーカイブ属性 185
 - アーカイブの場所 192
 - アクセスリストグループ権限 232
 - アクセスリストグループのメンバー 233
 - アクセスリストの定義 235
 - ユーザ ID 230
 - ユーザ権限 231
 - アクセスコントロールデータベースとの関連付け 236
- 変更許可、コンフィグレーションオプション 116
- 変更禁止、コンフィグレーションオプション 116
- 編集、nfsmap ファイル 43, 63
- ベース権限
 - 概要 206
- 別ラインによる開発 260

ほ

- 保存、デルタ 113
- 防止
 - アーカイブへの同時アクセス 108
 - データの破損 108

ま

- マージ
 - 概要 268
 - 自動 269
- マスク 113
- マスターコンフィグレーションファイル
 - Version Manager への埋め込み
 - コマンドラインインターフェイス 60, 119
 - デスクトップクライアントの場合 118
 - オプションの変更禁止 116
 - 概要 75, 80

む

- 無効化、setuid 63
- 無効化、セキュリティ 235

へ

- 並行開発のパス 252, 266

め

- メニュー項目イベント 274
- メニューバー、カスタマイズ 182

ゆ

- ユーザ 202
 - ID の変更 230
 - アクセスコントロールデータベースからの削除 230
 - 権限の割り当て
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 216
 - コマンドラインインターフェイスでの構文 224
 - 自動的に作成 214
 - 定義
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 214
 - 定義の複製 217
 - パスワード 215, 217
 - 有効期日 215
- ユーザ ID 90
 - 大文字と小文字の区別
 - コマンドラインインターフェイス 66
 - デスクトップクライアントの場合 50
 - 変更 230
- ユーザグループ 205
- ユーザのパスワード 215, 217
- ユーザのワークフロー 19
- 有効化
 - セキュリティ
 - コマンドラインインターフェイス 226
 - デスクトップクライアントの場合 221
- 有効期日、ユーザ 215

ら

- ライフサイクルの管理 251

り

- リナンバ 113
- リビジョン
 - 複数のアクセスの制御 86
 - ロック 86
 - ロックの解除 86
- 履歴レポート
 - 概要 297
 - ジャーナルレポートの読み方 298
 - 生成
 - コマンドラインインターフェイス 301
 - デスクトップクライアントの場合 299

る

- ルール
 - 権限 207
 - プロモーションモデル 246

れ

- レポート
 - オプションの設定 290
 - ジャーナルレポート
 - 概要 293
 - ジャーナルレポートの読み方 293
 - コマンドラインインターフェイスにおける生成 296
 - デスクトップクライアントにおける生成 294
 - セキュリティ
 - 概要 302
 - コマンドラインインターフェイスにおける生成 304
 - ジャーナルレポートの読み方 302
 - デスクトップクライアントにおける生成 304
 - フォーマット 290
 - フォーマットのカスタマイズ 292
 - 履歴
 - 概要 297
 - コマンドラインインターフェイスにおける生成 301
 - ジャーナルレポートの読み方 298
 - デスクトップクライアントにおける生成 299
- レポートの表示
 - 詳細なメッセージ 106
 - 著作権 106
 - バージョン情報 106

ろ

- ローカルコンフィグレーションファイル
 - 概要 79
 - 作成 58
- ログイン
 - シングルサインオン 91
 - ソース 90
 - ソース、設定 92
 - ソースの埋め込み 93
- ログインソース
 - Host Id 90
 - LDAP Id 90
 - Netware Id 91
 - VCS Id 91
 - WNet ID 91
 - 概要 90
 - ログインダイアログボックス 90
- ログインダイアログボックス

- 概要 90
- ユーザパスワードの指定 215, 217
- ログファイル
 - change.log 304
 - convERR.log 335
 - dgb_path.log 335
 - Shared.log 320, 335
 - ss2vm.log 334
- ロック 86
- ロック、複数 84

わ

- ワークスペース
 - 階層 175
 - 概要 174
 - 作成、削除、名称変更の制限 178
 - 自動ブランチ定義の例 262
 - 設定 181
 - 設定の表示 176
 - タイプ 174
 - パブリック
 - 概要 174
 - 作成 179
 - 使用 177
 - プライベートワークスペース 174
 - ルート 174
- ワークスペースの設定
 - 定義、デフォルトプロモーショングループ 174
- ワークフロー 19
- 割り当て
 - アクセスリストグループの権限
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 219
 - アクセスリストグループのメンバー
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 219
 - クロスプラットフォームで使用する権限
 - コマンドラインインターフェイス 64
 - デスクトップクライアントの場合 44
 - ユーザの権限
 - コマンドラインインターフェイス 224
 - デスクトップクライアントの場合 216