



比較： PERFORCE と Subversion

注：この資料は、Perforce Software 社が作成したドキュメントを東陽テクニカが日本語に翻訳したものです。
オリジナルは、http://www.perforce.com/perforce/comparisons/perforce_subversion.pdf をご参照ください。
この資料の中で説明されている機能のうち、日本語版 PERFORCE では未サポートの機能があります。
この使用に関してご不明な点は、東陽テクニカ技術サポート perforce@toyo.co.jp 宛にお問い合わせください。

概要

本書では、PERFORCE (バージョン 2008.1)と、オープンソースのソフトウェア構成管理(SCM)ツールである Subversion (バージョン 1.5.2)とを比較します。機能面での主要な相違点を対比し、ブランチ、マージ、チェックイン、チェックアウトなどの典型的な操作における一対一での評価基準を示すとともに、両システムでのスケーリングの効果を全般的に比較します。

機能概要

属性	PERFORCE	Subversion
ブランチ機能とマージ機能	PERFORCEでは、高度かつ充実したマージ追跡メカニズムにより、自動的にすべてのブランチ操作の履歴が追跡される。	Subversionはブランチ機能を提供しており、基本的なマージ履歴の記録をマージ先ファイルの属性に格納している。Subversionのマージ・アルゴリズムは、マージ・シナリオのすべてを網羅してはいない。
分散開発	PERFORCEプロキシにより、リモート・ユーザを対象に無制限に展開が可能なキャッシング・ソリューションが提供されている。最小限の管理作業しか発生せず、追加費用は必要としない。PERFORCEの中心的な製品エンジニアにより設計、開発、および社内での保守が行われる。	リモートのスレーブ・サーバを、中央のマスター・サーバから同期することが可能。同期処理を行うにはイベントベースのスクリプトを作成する必要がある。
拡張性とパフォーマンス	PERFORCEのアーキテクチャは何テラバイト、何千万個にも及ぶバージョン化ファイルに対応して拡張可能であることが証明されており、今後もそれが保証される。高速化をめざしたアーキテクチャにより、PERFORCEは直線的に拡張可能である。10,000サイトを超えるユーザ環境で実装されている。	Subversionは、CPUを集中的に使用する処理に大いに依存しており、ワークロードが増加すると拡張性が低くなる。
欠陥追跡	PERFORCEにはジョブと呼ばれる基本的な欠陥追跡システムが提供されており、主要なサードパーティの欠陥追跡システムとの統合が可能である。	Subversionでは提供されていない。別の欠陥追跡ソリューションを使用する必要がある。
関連ツールとの統合	PERFORCEはEclipseやVisual Studioなどの主要なIDE、ビルドツール、Microsoft Office、Adobe Photoshop、Maya、その他多くのサードパーティ製ツールとの統合が可能である。	統合ツールは、オープンソースコミュニティから入手可能である。
システム管理およびサポート	Perforce Software社の技術担当者が提供する。30万人を超えるユーザをサポートしている。	別料金で、サードパーティからサポートを受けることができる。

ブランチ機能とマージ機能

Subversion は、単純なコピーコマンドを使用してブランチを作成するブランチ機能を備えており、各ファイルに別々に添付されている基本的な操作履歴を、ファイルをキーにした属性リストの中に保存します。Subversion のマージ処理は低速で、従来から問題が多く、マージ・エラーが発生することがあります。兄弟ブランチや、遠く隔たって関連するブランチの間での一般的なマージ・シナリオが、実現できない場合も多くあります。そのようなシナリオでは、ユーザは操作を中止して手動でマージを行わなければなりません。双方向マージおよび不一致型マージは、サポートされていません。

PERFORCE のインターファイル・ブランチング™ モデルは強力かつ柔軟であり、ブランチとマージの完全な履歴を保持しながら、数千個のファイルを迅速にブランチ処理することが可能です。ブランチ間のすべての変更を手動で追跡する代わりに、ユーザは PERFORCE を利用して複数ブランチ間のファイルの変更を自動的に、または解決を要する衝突がより少ない状態でマージすることができます。リビジョン・グラフというグラフィカル・ツールは、各ファイルの詳細なブランチ履歴やコードの拡がりを、分かりやすくビジュアルに表示します。

PERFORCE は本質的に、ブランチ間で共有される、祖先の関連ファイルを把握しています。変更が反映元ブランチから反映先ブランチへと反映されれば、変更は両方のブランチに確実に存在することになります。したがって、それ以降の反映先ブランチから反映元ブランチへの逆方向の反映において PERFORCE は、反映先ブランチに直接加えられた変更だけを反映するよう、ユーザに要求します。このような反映様式は、双方向マージまたははしご型マージと呼ばれます。

不一致型ブランチ処理は、反映元ブランチに直接導入された変更または特性を、反映先ブランチへのマージの際に無視またはスキップするというケースに対応します。実際のところ、反映の記録は作成されますが、反映先ファイルは変更されません。このような反映様式は、PERFORCE では Ignore (無視) レコードとして知られています。反映履歴では保留中のマージがないことが示しつつ、反映元ブランチと反映先ブランチは異なった状態を保持します。

分散開発

Subversion では、リモート環境の開発者は直接ローカルのスレーブ・サーバに対して作業を行います。すべてのスレーブ・サーバは、中央のマスター・サーバから格納データを同期するように構成されています。マスターでの更新処理によって起動したイベントベースのスクリプトが、登録されたリモート・スレーブ・サーバのそれぞれに対し、アーカイブを同期させるように通知します。スクリプトは、スレーブ・サーバとマスター・サーバの両方に作成し、配置する必要があります。

PERFORCE を使用して分散開発する場合でも、余分なプロセス・オーバーヘッドや追加費用を必要とすることはありません。リモート環境の PERFORCE プロキシが、PERFORCE の分散アーキテクチャをサポートします。PERFORCE プロキシは、リモート環境にいるユーザにファイルをキャッシュして提供するため、低速な WAN リンクにおけるトラフィックを減少させます。ローカル環境またはリモート環境のすべてのユーザが、中央にある単一の PERFORCE データベースに接続することにより、まったく同じプロジェクト・ファイルを参照することができます。任意の数のプロキシを、エンドユーザからは不可視の状態に配置することができ、最小限の管理作業しか必要としません。

拡張性およびパフォーマンス

Subversion は、バイナリ差分比較方式を使用してバイナリファイルおよびテキストファイルをバージョン化するため、クライアントおよびサーバの CPU 時間を非常に多く費やします。このことはファイルが大きい場合や、同じリソースを取り合う同時実行ユーザが多い場合に、特に問題となります。

PERFORCE は、速さを最優先して設計されています。テキストファイルはリバース・デルタ (RCS) 形式、バイナリファイルは GZIP 圧縮形式で保存するため、ユーザはチェックインやその他の標準的なソフトウェア構成管理 (SCM) 操作を、すぐに実行することができます。PERFORCE は直線的に拡張されるため、特定のファイルのリビジョン数やファイル・サイズの影響で、パフォーマンスが低下することはありません。PERFORCE サーバは、数千ユーザを擁する環境での実装に成功しています。

ベンチマーク

比較のためにベンチマーク・テストを行ったところ、すべての同等の SCM 操作において PERFORCE は、Subversion よりはるかに優れているという結果を得ました。詳細な試験結果を以下に示します。

テストデータ

テストを実行したクライアント/サーバ・マシンの組み合わせに関して、仕様は以下のとおりです。

- Opteron246、シングル・コア・プロセッサ、2 GHz で稼動
- メモリ総量: 8GB
- オペレーティングシステム: CentOS 4.3
- ストレージ: Standard Maxtor DiamondMax SATA2 を使用する 8 ウェイ RAID10

テスト結果

PERFORCE と Subversion のベンチマークを比較した結果を、以下の表に示します。応答時間は分、秒の単位で測定されています。

下に示す表 1 のデータセットは、24,353 個のファイルから構成され、約 327,292 MB のディスク容量を消費する完全な Linux カーネルです。

下に示す表 2 のデータセットは、Linux カーネル・ツリーの 5 個のコピーから構成されています。このデータセットは、全部で 121,767 個のファイルを持ち、合計サイズは 1.6 GB です。

表1: 完全なLinuxカーネルを1つコピーすることによる性能測定結果

操作	PERFORCE (2008.1)	Subversion (1.5.2)
完全なLinuxカーネルの追加(データは上述)	1分6.681秒	1分8.265秒
カーネルツリー全体の小規模なブランチ処理(*1)	0分3.940秒	0分0.072秒
ブランチをデスクトップに同期	0分12.382秒	0分58.054秒
カーネルツリー全体の大規模なブランチ処理(*2)	0分11.062秒	2分7.131秒
編集ファイル611個、新規ファイル15個におけるファイル変更を検出	0分5.605秒	0分20.152秒
編集ファイル611個、新規ファイル15個をサブミット	0分2.688秒	0分52.855秒
7個のAVIファイル(合計サイズ1.3GB)をチェックイン	2分22.042秒	16分5.675秒
7個のAVIファイル(合計サイズ1.3GB)をデスクトップに同期	0分22.073秒	6分10.005秒
2個のバイナリファイル(合計サイズ800MB)への変更をチェックイン	1分18.601秒	0分44.949秒
バイナリファイルへの大規模な変更をチェックイン	0分59.650秒	1分21.450秒
611個の変更ファイルを子ブランチに自動マージ(マージとチェックイン)	0分10.686秒	1分12.297秒
611個の変更ファイルを子ブランチから兄弟ブランチへと自動マージ(マージとチェックイン)	0分8.918秒	0分43.265秒

(*1)「小規模なブランチ処理」では、サーバ上のファイルをブランチするのみ。ファイルは一切クライアントに転送されません。

(*2)「大規模なブランチ処理」では、サーバ上のすべてのファイルをブランチした後、それらをクライアントに転送します。

表2: Linuxカーネルツリーを5つコピーすることによる性能測定結果

操作	PERFORCE (2008.1)	Subversion (1.5.2)
5つのカーネルツリーすべてをソースコントロールに追加	7分38.155秒	118分10.367秒
ツリー全体の小規模なブランチ処理(*1)	0分26.498秒	0分3.065秒
ツリー全体の大規模なブランチ処理(*2)	2分53.466秒	53分2.597秒
121,767個のファイルをデスクトップに同期(1.6GB)	1分51.649秒	9分31.325秒

(*1)「小規模なブランチ処理」では、サーバ上のファイルをブランチするのみ。ファイルは一切クライアントに転送されません。

(*2)「大規模なブランチ処理」では、サーバ上のすべてのファイルをブランチした後、それらをクライアントに転送します。

欠陥追跡

Subversion には欠陥追跡機能が含まれていませんが、少数のサードパーティ製統合ツールが利用可能です。

PERFORCE では、ジョブと呼ばれる基本的な欠陥追跡システムが提供されています。通常、ジョブは改善要求または修正すべきバグを表します。ジョブの定義をカスタマイズして、PERFORCE のワークフローに対応させることができます。また、ジョブは複数のサードパーティ製欠陥追跡システムとの統合が可能です。

PERFORCE のジョブに入力されたデータは自動的に欠陥追跡ツールに複製され、またその逆にも複製されます。PERFORCE は、そのバグ修正が最初に組み込まれたブランチに関係なく、特定のブランチにバグ修正が存在するかどうかを示すことができます。

関連ツールとの統合

オープンソースコミュニティでは、Subversion のさまざまな GUI、および主要な IDE で動作するプラグインをサポートしています。

PERFORCE は洗練されたマルチプラットフォーム GUI を備えており、主要な IDE のためのプラグインが提供されています。また PERFORCE は、次のような他の代表的なアプリケーションとの統合が可能です。

- IDE
- ウェブおよびグラフィック・ツール
- ソフトウェアのビルドツール
- Microsoft Office
- API ツール
- マージおよび比較ツール

システム管理およびサポート

Subversion では、実装の前に管理者が決定しなければならない項目があります。例えば管理者は、ユーザ認証のために Web サーバを構成するか、ローカルのテキストベースの構成ファイルによるメカニズムを使用するかを決定しなければなりません。また、バージョン化ファイルの保存のために、ファイルベースのシステムを使用するかデータベースを使用するかを、決定する必要があります。

Subversion はソースコードに完全にアクセスできるようになっていますが、管理者は時間と労力をかけてコードベースに慣れなければなりません。Subversion には、サードパーティによる有償サポートがあります。

PERFORCE に組み込まれたセキュリティ・メカニズムは、プラットフォームにかかわらず同様に機能し、多くのアクセス・レベルを提供します。このメカニズムは、個々のファイルからディポ全体に至るまで、自由な範囲のバージョン化ファイルに対して適用することができます。バージョン化ファイルは、すべてのプラットフォームで同様に保存されます。

PERFORCE の技術サポートは、高度な技術と早い応答を特徴としており、製品の評価中であってもそ

のサポートを受けることができます。Perforce Software 社では、お客様が SCM システムを購入されるとき、ソフトウェアを信頼するだけではなく、技術サポートが受けられることも期待していると考えています。

その他のリソース

PERFORCE を評価する

約 5,000 の企業の 30 万人を超えるユーザに、ソフトウェア構成管理ツールとして PERFORCE をご利用いただいています。Perforce Software 社は、購入を検討しているお客様ご自身で判断していただくため、通常 45 日間の試用評価をお勧めしています。技術サポート部門の品質については、説明を読むよりも経験していただく方が得策です。まずは、<http://www.perforce.com/svn> にアクセスしてください。

PERFORCE のデモを依頼する

PERFORCE に関してさらにお知りになりたい場合は、お客様の条件に適合させた対話型のデモを依頼してください。<http://www.perforce.com/svndemo> にアクセスしてください。

PERFORCE への移行

Subversion から PERFORCE へと移行するための変換ユーティリティを含む、関連ソフトウェアの詳細情報は、<http://www.perforce.com/perforce/loadsupp.html#conv> にあります。

Perforce コンサルティング・サービスは、顧客の様々な SCM システムからの移行作業を支援した経験があります。詳しくは、<http://www.perforce.com/consulting> にアクセスしてください。

その他の SCM との比較記事を、<http://www.perforce.com/comparisons> に掲載しています。

【お問合せ先】

株式会社東陽テクニカ ソフトウェア・ソリューション

TEL 03-3245-1248 FAX 03-3246-0645

URL <http://www.toyo.co.jp/>

営業的なお問い合わせ ss_sales@toyo.co.jp

技術的なお問い合わせ perforce@toyo.co.jp

