

Title	オープンソースソフトウェア開発に適したリポジトリ分散の支援機構に関する研究
Author(s)	中島, 健至
Citation	
Issue Date	2005-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1925
Rights	
Description	Supervisor:落水 浩一郎, 情報科学研究科, 修士

オープンソースソフトウェア開発に適した リポジトリ分散の支援機構に関する研究

中島 健至 (310076)

北陸先端科学技術大学院大学 情報科学研究科

2005年2月10日

キーワード: オープンソースソフトウェア開発, 管理方針, リポジトリ, 構成管理システム, CVS.

背景と目的

ネットワーク技術の発達により, 広域分散環境においてソフトウェアの共同開発が行われている. その1つの形態に, オープンソースソフトウェア開発 (OSSD) がある. OSSDでは, 規模が大きくなると複数の子プロジェクトを持つことがある. これらは, ネットワーク的に遠く離れ, プロジェクトの管理方針が大きく異なっている. このようなプロジェクトにおいて成果物と変更履歴を管理する1つのリポジトリを用いることは困難である. そこで, リポジトリの分散構成が必要となる.

既存の構成管理システムや外部ツールの分散機能では, 必要以上に成果物と変更履歴を転送するため, 開発プロジェクトに応じて柔軟なディレクトリ構造やリビジョンツリー構造をとることが出来ない. また, 転送の方向が単方向で子から親へ反映できない場合がある. そのため, 既存の OSSD 見られる多様なリポジトリの分散構成をサポートできない.

そこで本研究では, 構成の異なるリポジトリ間で成果物と変更履歴を必要な粒度で転送する機構を元に, OSSD を適切に支援可能なリポジトリ分散の支援機構を実現する. 本研究では, OSSD で広く使用されている CVS を対象としてクライアントシステムの設計と実装を行う. 本研究が提案する機構により, 開発プロジェクトは, CVS の変更や専用のサーバプログラムを必要とせず, 独自のリポジトリの構築と運用ができる.

リポジトリ分散の支援機構

本研究が実現する機構は次の特徴を持つ。

- 任意の粒度のデータ転送
任意の粒度においてリポジトリ間の双方向でデータ転送を可能とする。粒度とはリビジョンから、ファイル、ディレクトリ、リポジトリまでのことを指す。
- 管理方針の分離
転送元のリポジトリの構成、つまりディレクトリ構造、ファイル名、リビジョンツリー構造に制約を受けず、これらを元に転送先で独自の構成のリポジトリを構築可能とする。

これらを、細粒度の操作とデータ転送管理部により実現する。細粒度の操作は転送元のリビジョンツリーから差分またはファイル全体を取得して転送先に1つのリビジョンを作成する。この細粒度の操作により転送元のリポジトリと異なるリビジョンツリーを構成できる。データ転送管理部によりリポジトリ間のファイル名の対応やディレクトリ構造の対応を行う。また、細粒度の操作、上位操作、ディレクトリの作成を繰り返し実行することで、リビジョン単位からリポジトリ全体までのデータ転送を処理する。

さらに、OSSDにおいて一般的に必要な以下のリビジョンツリーの上位操作を細粒度の操作を用いて記述した。

- graft リビジョンツリーの一部を転送する
- replicate リビジョンツリー全体を複製する
- add トランクまたはあるブランチの開発進展部分を転送する
- add_tree リビジョンツリー全体の開発進展部分を転送する

本研究の有効性

FreeBSDなどの多くのプロジェクトでリポジトリの分散に用いられているCVSupの以下の2つの動作モードを細粒度の操作の組み合わせにより、記述できることを示した。

- exact 他拠点のリビジョンツリーと一致する
- non_exact 独自のブランチを維持したまま、他拠点の差分を取り込む

さらに、本研究の機構を用いた場合、CVSupを用いる開発者に開発の自由度、双方向の転送を提供できる例を示した。

従来のリポジトリ分散支援機構では支援の困難な2つの例、CygwinとKAMEの両プロジェクトに必要なリポジトリの部分複製、子から親への転送の支援ができることを示した。

まとめと今後の課題

本研究では、独自のリポジトリ構造を持つ OSSD に対し適切な支援を行う、CVS リポジトリの分散支援機構の設計と実装を行った。これは任意の粒度のデータ転送と管理方針の分離という特徴を持つ。これらにより、プロジェクトに応じたリポジトリの構築と運用が可能となる。各拠点のプロジェクトは他拠点と関連を維持しつつ、独自の管理方針に基づいてリポジトリを運用できるので、開発効率の向上が期待できる。

今後の課題として、異なる構成管理システム間のデータ転送と協調支援機構の実現が挙げられる。例えば、CVS-Perforce 間や CVS-Bitkeeper 間などでデータ転送を可能とすることにより、拠点間で異なる構成管理システムを用いた開発を行える。本研究では拠点間で独自のリポジトリ構成を持つことを可能としたが、拠点間で矛盾が生じることを防ぐ機構を提供していない。そこで、協調支援機構として、時間的なリポジトリ間のずれを解消する即時反映や転送ミスを防ぐ矛盾防止機構の開発が必要となる。