

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	<a href="http://hdl.handle.net/10119/1925">http://hdl.handle.net/10119/1925</a>
Rights	
Description	Supervisor:落水 浩一郎, 情報科学研究科, 修士

# Repository Decentralization Support Mechanism for Open Source Software Development

Takeshi Nakashima (310076)

School of Information Science,  
Japan Advanced Institute of Science and Technology

February 10, 2005

**Keywords:** Open Source Software Development, Repository, Management Policy, Software Configuration Management Systems, CVS.

## Background and Purpose

Cooperative development of software in wide area distributed environment has been performed by the evolution of the Internet and the improvement of computer resources. One of the developments is open source software development. In this development project, it has two or more subprojects in many cases when the scale grows like Mozilla, Apache and so on.

These subprojects far from the network. The management policy of them might be greatly different. When the project with different purpose exists together mutually in one repository, the management policy of the main project is compelled to the subprojects. To evade this, the repository that manages the result thing and the change history should be able to adopt a decentralized composition.

In existing configuration management systems and external tools, there are somethings with a decentralized function of the repository. However, because the result thing and the change history are forwarded in these configuration management system and external tool more than the necessity, a flexible directory structure and the revision tree structure cannot be taken according to the development project. Moreover, the direction of forwarding might not be able to be reflected from the subproject to main project in a single direction. Therefore, it is difficult to support the development projects such as Cygwin and KAME that do construction and the operation of an original repository that exists in a open source software development.

Then, in this research the support mechanism of the repository decentralization that can appropriately support a open source software development is achieved based on the mechanism that forwards the result thing and the change history between repositories with a different composition by a necessary granularity. This research, design and implement the client system for CVS widely used by a open source software development. Neither any change of CVS nor any additional server program are needed, and the development project can be operated with the construction of an original repository with the mechanism that this research proposes.

## Support mechanism of repository decentralization

The mechanism that achieves this research has the following feature.

- **Data transfer of arbitrary granularity**

It support mutual data transfer in an arbitrary granularity. The granularity indicates revision, the file, the directory, and the entire repository here.

- **Separation of management policy**

It allow the destination repository to have a different structure from the source. The destination can be different in its structure of directories, filenames, or structure of a revision tree.

These are achieved by the fine grained operation and the data transfer management mechanism. Fine grained operation acquires the difference or the entire file from revision tree in the source repository and makes one revision at the destination. A revision tree different from the repository in the source according to fine grained operation can be composed. The correspondence of the file name between repositories and the directory structure correspond by data transfer management mechanism. Data transfer management mechanism processes the data transfer from the unit of revision to the entire repository by repeatedly executing an operation of the minute grain degree or a high-level operation and making the directory.

In addition, a high-level operation of necessary following revision tree was generally described by using fine grained operation in open source software development.

- **graft** It is grafted tree partially of revision tree.
- **replicate** The entire revision tree is reproduced.
- **add** The development progress part of the trunk or a certain branch is forwarded.
- **add\_tree** The development progress part of the entire revision tree is forwarded.

## Effectiveness of this research

It was shown to be able to describe two operation modes of the following of CVSup used to distribute the repository by a lot of projects such as FreeBSD by combining the operations of the minute granularity.

- **exact** It agrees to revision tree of another base.
- **non\_exact** The difference of another base is taken with an original branch maintained.

In addition, when the mechanism of this research was used, the example in which forwarding the degree of freedom of development and interactive was able to be offered to the developer who used CVSup was shown.

It was shown to be able to support forwarding from a partial reproduction and the sub-project to main project of a necessary repository for both projects of 2 difficult support examples and Cygwin and KAME in a past repository decentralization support mechanism.

## Conclusion and Future Works

In this research, the design and implement the decentralized support mechanism of the CVS repository that did appropriate support to the open software development project with an original repository structure were done. This has the feature of separation of the data transfer and the management policy of an arbitrary granularity. Construction and the operation of the repository according to the project become possible by these. Because the project of each base can operate the repository based on an original management policy maintaining another base and the relation, the improvement of the development efficiency can be expected.

The different data transfer between the configuration management systems and the achievement of cooperated support mechanism are enumerated as a problem in the future. For instance, it is possible to develop by enabling the data transfer between bases between CVS-Perforce and CVS-Bitkeeper etc. the departure for a different configuration management system.

The mechanism that prevents contradiction being caused between bases is not being offered though an original repository composition was enabled to be had between bases in this research. Then, the development of the contradiction prevention mechanism that prevents a reflection and an immediate forwarding mistake that cancels the gap between time repositories as cooperated support mechanism is needed.

Development advances as the hours go by. The gap is generated in the result thing and the change history between repositories. The reflection cancels the gap between time repositories at once because it immediately reflects the result thing and the change history between repositories if committed to cancel these.

The operation mistake of the data transfer might be done because there is a limit in man's acknowledgment ability when the number of the directory and file increases. The contradiction prevention mechanism increases useless revision by it, and prevents an increase of the file and the conflict, etc. being generated. The following methods are devised now. It is made not to forward as a rule as a prohibition condition of the directory and the file of the base a set enabling. It is enabled that it is specification of the argument of the repository of another base, the directory name, the file name, the operation name, and the operation as the exception condition. As a result, the file of the base can forward, and, for instance, do only the file that exists in a certain directory and forwarding only as a certain repository. Moreover, various support like the agreement etc. of the agreement of the entire revision tree, the agreement of the trunk and a certain branch, certain revision, and a certain revision can be done by specifying the argument of the operation name and the operation.