



Network-Unified Configuration Management

Evolutionary Design of Complex Software

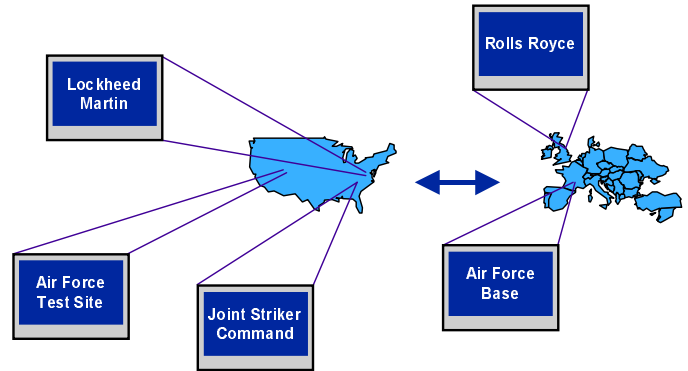
Software Engineering Research Laboratory

University of Colorado at Boulder

NUCM is a generic, peer-to-peer repository supporting distributed Configuration Management (CM). Its programmatic interface allows for the rapid construction and evolution of CM systems, whereas its underlying distribution mechanism facilitates Configuration Management in the context of large-scale, wide-area software development.

NUCM separates CM *repositories*, which are the stores for versions of software artifacts and information about these artifacts, from CM *policies*, which are the specific procedures for creating, evolving, and assembling versions of artifacts maintained in the repository. Combined, a CM repository and a CM policy comprise a complete CM system. But, it is their separation into two architectural components that, through reuse of the NUCM CM repository, facilitates the rapid development of complete CM systems.

With NUCM's generic programmatic interface it becomes feasible to develop a CM system that specifically supports and is tailored to an organization's internal software development process and policies. Until now, an organization was forced to buy a commercial CM system and adopt the process and policies incorporated in the acquired CM system. NUCM reverses this approach and instead allows the CM system to be specialized to the actual process and policies taking place.

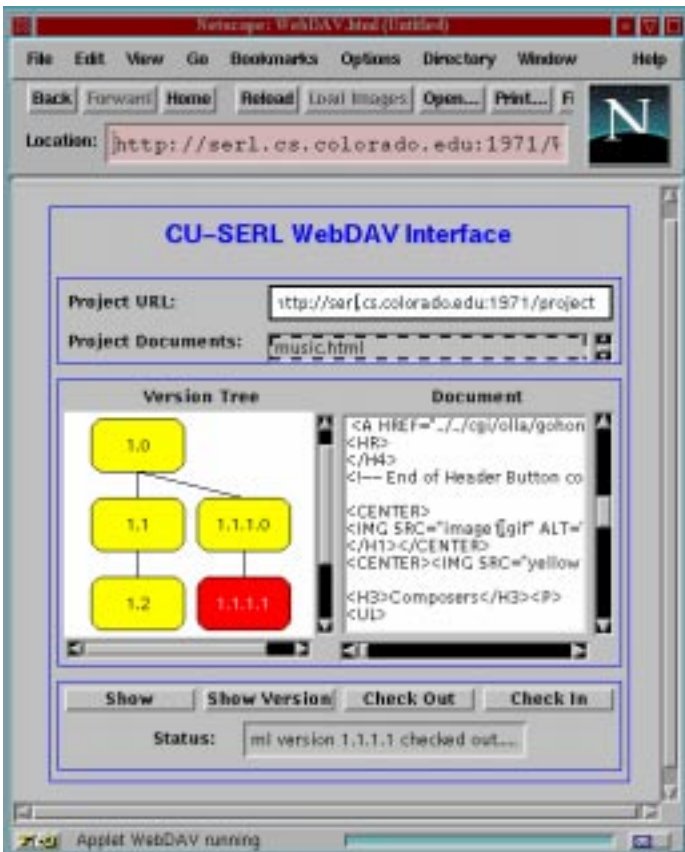


Distributed and decentralized Configuration Management in such settings as depicted here is a problem that has not been solved as of today.

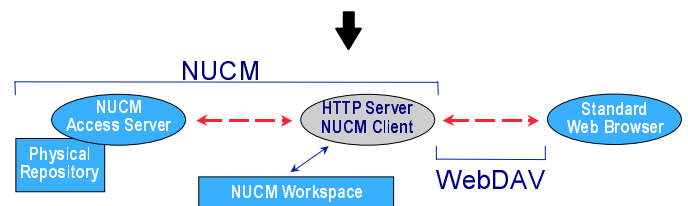
Benefits

NUCM provides the following benefits to a CM system developer:

- **Rapid development.** NUCM's reusable CM repository, combined with its generic interface, allows for the rapid construction of complete CM systems.
- **Distributed operation.** Any CM system developed with NUCM inherits NUCM's distributed nature, and can have CM clients and servers spread across the world.
- **Scalability.** NUCM's peer-to-peer architecture, combined with its lightweight implementation, presents a CM system developer with a scalable repository capable of operating in wide-area, large-scale Inter- and Intranets.
- **Flexibility.** The NUCM programmatic interface is generic, and supports the creation of a wide variety of CM policies.
- **Type independence.** NUCM can store and version any type of artifact.
- **Evolvability.** The NUCM repository supports the controlled evolution of artifacts through its versioning interface.



This example presents a prototype implementation of the new IETF Web Distributed Authoring and Versioning (WebDAV) standard, which prescribes use of the check-in/check-out CM policy. NUCM allowed this system to be constructed in less than a week and with only a few hundred lines of C and Java source code.



Data Model

The data model of NUCM is based on a flexible grouping mechanism in which atoms (individually versioned artifacts) and collections (groups of versioned artifacts) are treated identically. The data model maps naturally into the file system so that existing tools can manipulate the artifacts in their native environment. Furthermore, it is policy independent, and does not imply any relationship among the versions of an artifact.

Distribution Model

NUCM provides the concepts of *physical* and *logical* repositories. A physical repository is the actual store for some set of artifacts at a particular site. A logical repository is a group of one or more repositories acting as a single repository. CM policies interact with a logical repository and can therefore manipulate any of the artifacts irrespective of physical location. Many different distribution topologies can be modeled by NUCM, such as client-server or peer-to-peer. NUCM physical repositories and CM policies can be distributed throughout the world, while all are part of a single CM system.

Generic Programmatic Interface

NUCM's programmatic interface supports CM system developers with a policy programming language. For example, the familiar check-in/check-out policy reduces to:

check-out: open + testandsetattribute + initiatechange

check-in: commitchange + removeattribute

This simplicity is intrinsic to NUCM; its interface functions have been carefully tuned to be simple yet powerful

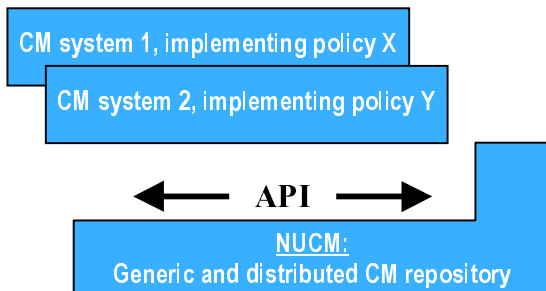
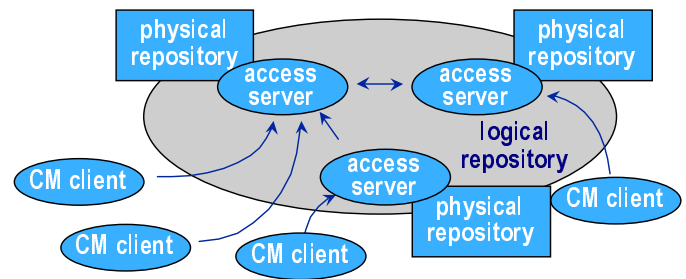


Illustration of the NUCM architecture. The NUCM generic and distributed CM repository is reused in the construction of multiple CM systems.



Three NUCM access servers combine to provide CM clients with a single distributed repository of versioned artifacts.

Experience

NUCM is in use in two systems that are publicly available, SRM and DVS, as well as one experimental system, WebDAV. Both the development time and development effort of these systems were greatly reduced due to the use of NUCM. For example, DVS is a fully functional, distributed versioning system that required only 1500 new lines of C source code.

Availability and Support

The current version of NUCM, *NUCM-2.1*, is freely available software. It is written in C, and is supported on most Unix platforms, including Solaris, Sun4, HP-UX, and Dec Alpha. A new, Java-based version of NUCM is currently being prepared.

Future Work

Future work on the NUCM project will focus on two areas. First, we will develop standard CM policy libraries, which, through additional reuse besides the NUCM CM repository, will even further reduce the effort of implementing a CM system. Second, we will investigate the problem of CM policy integration in order to support a set of autonomous organizations to join in cooperative development.

References

- [1] A. van der Hoek, D. Heimbigner, and A.L. Wolf **A Generic, Peer-to-Peer Repository for Distributed Configuration Management**. In *Proceedings of the 18th International Conference on Software Engineering*, Berlin, Germany, 1996. Available on the Web at: <ftp://ftp.cs.colorado.edu/users/andre/papers/ICSE18.ps>.
- [2] IETF WebDAV Working Group **World Wide Web Distributed Authoring and Versioning**. Available on the Web at: <http://www.ics.uci.edu/~ejw/authoring/>.

Principal Investigators

Dennis Heimbigner
Alexander L. Wolf
Department of Computer Science
Campus Box 430
University of Colorado
Boulder, CO 80309

{dennis, alw}@cs.colorado.edu
303-492-6643, 5263
303-492-2844 (fax)

Freely Available Software

Information and publications about the NUCM project, as well as its software, is freely available on the Web:

<http://www.cs.colorado.edu/serl/cm>

For further information, please contact:

André van der Hoek
andre@cs.colorado.edu

Acknowledgements

This work was supported in part by the Air Force Materiel Command, Rome Laboratory, and the Defense Advanced Research Projects Agency under Contract Numbers F30602-94-C-0253 and F30602-98-2-0163. The content of the information does not necessarily reflect the position or the policy of the U.S. Government and no official endorsement should be inferred.