Revision Control System

The RCS made its appearance in 1982. A revision control system (RCS) is an application capable of storing, logging, identifying, merging or identifying information related to the revision of software, application documentation, papers or forms. Most revision control systems store this information with the help of a differential utility for documents.

A revision control system is an essential tool for an organization with multi-developer tasks or projects, as it is capable of identifying issues and bugs and of retrieving an earlier working version of an application or document whenever required.

A revision control system is also known as a version control system.

Architechture :-

Repository Architectures

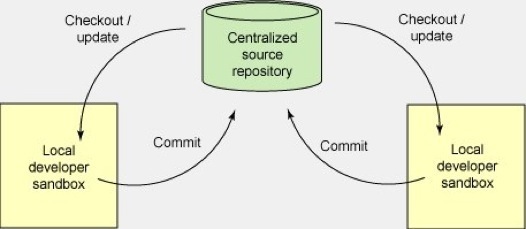
All versioning systems are based on one of two types of repository architectures:

* Client-server/Centralised Architecture
* Distributed /De-centralised Architecture

Client-server/Centralised Architecture

The historical and most common architecture found today is the centralized repository. In this architecture, users (clients) connect to a central source(server) for access to the repository.

Essentially “Developers check out source code from the central repository into a local sandbox and, after making changes, commit it back to the central repository”

http://www.cs.nott.ac.uk/~pszcah/G53QAT/Report08/too06u-WebPage/Site/Architectures_files/stroke.pnghttp://www.cs.nott.ac.uk/~pszcah/G53QAT/Report08/too06u-WebPage/Site/Architectures_files/stroke_1.png

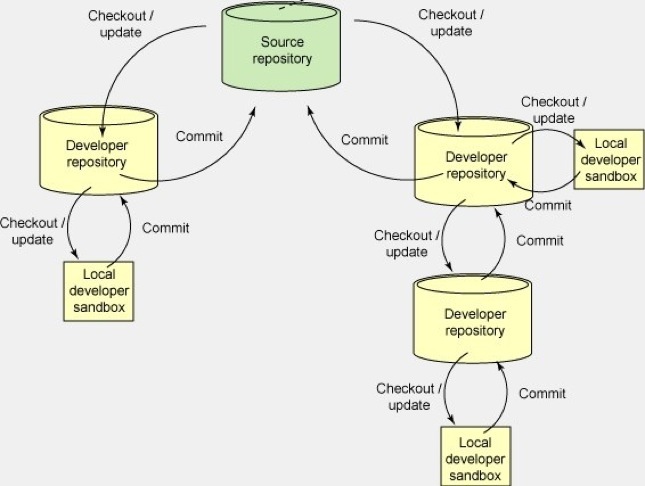
Branching is a process where subsections from the main source code are developed independently and then reassembled back when complete. Branching allows developers to edit section (branches) of the repository without posing major changes to the main source and retaining the revision history of the files.

Branches can be implemented in this architecture but not as efficiently as the Distributed Architecture.

**Distributed /Decentralised Architecture**

This architecture takes a peer-to-peer approach where each peer has a local repository which is similar to the original source repository to which they can commit their changes.

The main difference is that each clone is a full-fledged repository not dependent on network access or a central server.. Synchronization is conducted by exchanging patches (changesets) via peer to peer, making it possible for independent developers to work asynchronously.



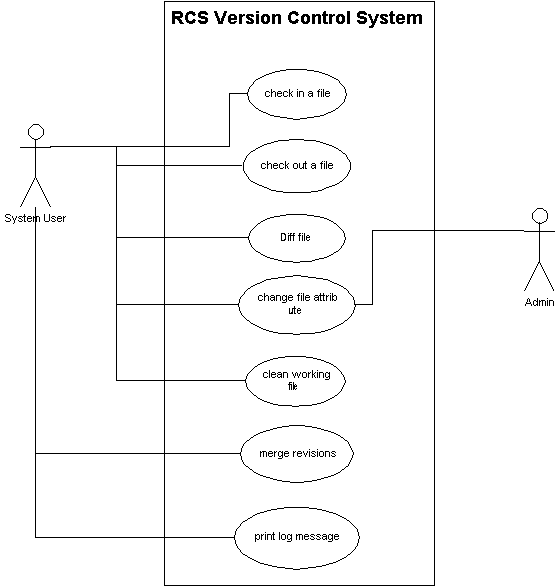
This is the model for many open source systems today, including the Linux® kernel

Since Linus Torvalds  gave a presentation at Google about a distributed version control systems called GIT in May 2007, the adoption and interest for Distributed Version Control Systems has been constantly rising.

Working of rcs

**There is not much to say about RCS. Its two most important tools ci and co can be used as usual. Note that RCS follows a "pessimistic" approach, which means only one person can have a lock on a file and is able to edit it, while other approaches to edit it are denied. So you should release your locks immediately after your changes. There are manual pages available on all RCS tools.**

**Diagram of rcs**



**Technical background**

**Most revision control systems run as independent standalone applications. There are two types of revision control systems: centralized and decentralized. Some applications like spreadsheets and word processors have built-in revision control mechanisms. Designers and developers at times use revision control for maintaining the documentation along with the configuration files for their developments. High-quality documentation and products are possible with the proper use of revision control systems.**

**https://www.techopedia.com/definition/30666/revision-control-system-rcs**