ONLINE SUBVERSION WEB CLIENT

KAVURI ROSHAN [1], KRANTHI KUMAR [2], MANOGNYA [3], LAVANYA [4], MOUNIKA [5].

[1,2,3,4&5]Department of Information Technology, JB.Institute of engineering & Technology Moinabad, Hyderabad, Andhra Pradesh, Imdia.
roshan.kavuri@gmail.com, kranthikumar.kandi@gmail.com

ABSTRACT

Subversion is a version control, source control or software configuration management (SCM), is the management of changes to documents, programs, and other information stored as computer files. It is most commonly used in software development, where a team of people may change the same files. As of now, there are lot of command based and stand alone based implementation of subversion tools are available like TortiseSVN RapidSVN [11]. But in the existing systems, data has to be maintained on each client system. This occupies large memory space on each client system. Instead of this, a web based implementation of subversion is being proposed in which data will be stored in centralized server rather than a local system .Here, each client is provided with some workspace in the server where the client maintains all his files in the workspace in different repositories. The client can even maintain his repositories which are present in different servers in the same server where his workspace is allotted.

Keywords

Subversion, repository, commit, update, add, delete, log, create, diff, checkout.

1. INTRODUCTION

1.1 Introduction to Subversion Web Client

Subversion is a free/open source version control system. That is, Subversion manages files and directories, and the changes made to them, over time. This allows you to recover older versions of your data or examine the history of how your data changed. In this Regard, many people think of a version control system as a sort of "time machine [1]. Subversion can operate across networks, which allows it to be used by people on different computers. At some level, the ability for various people to modify and manage the same set of data from their respective locations fosters collaboration[2].And because the work is versioned, you need not fear that quality is the trade-off for losing that conduit—if some incorrect change is made to the data, just undo that change. Subversion exists to be universally recognized and adopted as an open-source, centralized version control system characterized by its reliability as a safe haven for valuable data; the simplicity of its model and usage; and its ability to support the needs of a wide variety of users and projects, from individuals to large-scale enterprise operations.

Some version control systems are also software configuration management (SCM) systems. It is a general system that can be used to manage any collection of files. For you, those files might be source code—for others, anything from grocery shopping lists to digital video mix downs and

Natarajan Meghanathan, et al. (Eds): SIPM, FCST, ITCA, WSE, ACSIT, CS & IT 06, pp. 145–152, 2012.
© CS & IT-CSCP 2012
DOI : 10.5121/csit.2012.2315

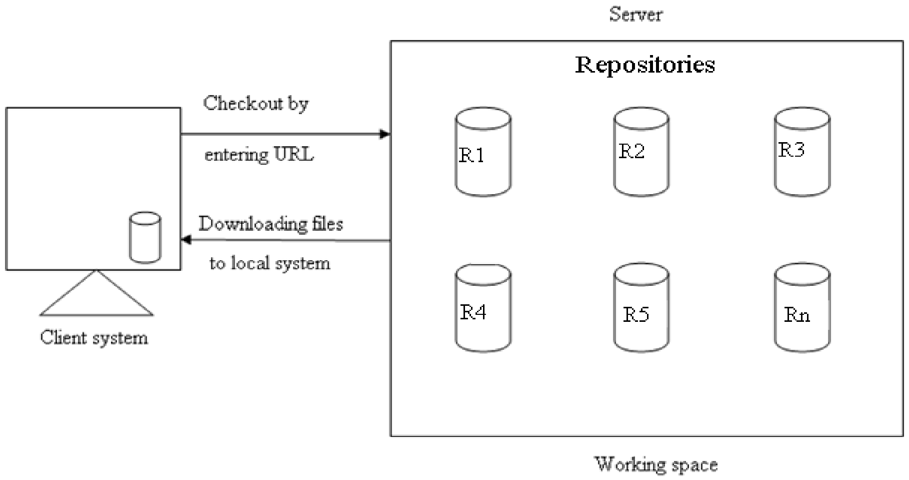
beyond. Developers use Subversion to maintain current and historical versions of files such as source code, web pages, and documentation [3]. Its goal is to be a mostly-compatible successor to the widely used Concurrent Versions System (CVS).

1.2 Existing System

There are allot of command based and stand alone based implementation of subversion tools available. But in all these existing systems, the client has to maintain a working copy of the repository and each time he want to view the changes made by other users of the repository, he needs to perform a checkout action due to which a working copy is again downloaded onto the client system [5].

Disadvantages:-

1. Each user has to maintain a working copy of the repository.
2. Also takes a large amount of time for obtaining the working copy during checkout option.
3. The user has to update his working copy, operate on it and then commit the changes to the server



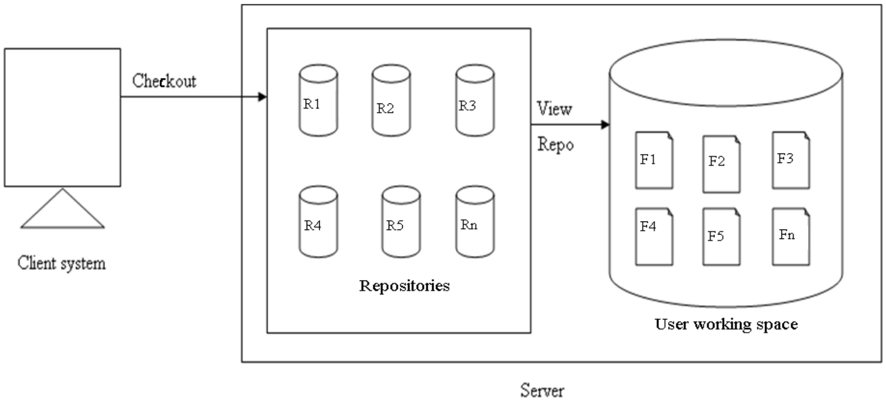
1.3 Proposed System

In this project we provide the clients with a web based implementation of subversion and to store the client data on the centralized server rather than on the local systems. Because whenever a client wants to view what changes has been made to the files in the repository, he has to download and view them for example Tortoises [11] as shown in above figure.

To make an SVN repository available to access using the HTTP protocol, you must install & configure web server at the server side and clients need not to install any thing except having uninterrupted web connectivity. Apache 2 is proven to work with SVN.

Advantages:

1. Each user need not maintain a working copy of the repository.
2. Memory on the user’s local system is saved.
3. Here the user can operate and commit the changes to the repository in the server itself.



2. MODULES

2.1. User Module:

The application starts by the user getting registered with the site. Once he is registered, he cannot directly checkout the repositories. The user needs to wait, until he is approved by the Admin. The Admin has to check the list of registered users, approve them. Once the Admin approves the users, an email is sent to the users registered mail id, informing him about the status, if he is approved or rejected. If he is approved, his user id and password are emailed. And a workspace is being allotted for the user in the server, where he can maintain his repositories. Making use of this user id and password, the user has to login.

Once the user logins, he can find few options like Repositories List, Checkout and Logout. Initially as the user logins for the first time, there are no repositories. He needs to perform a Checkout to obtain a working copy of the existing repositories into his workspace. For this, he needs to no the URL of the repository, username and password of the repository [7] [4]. The user can even maintain a working copy of the repositories present in different servers into a single server.

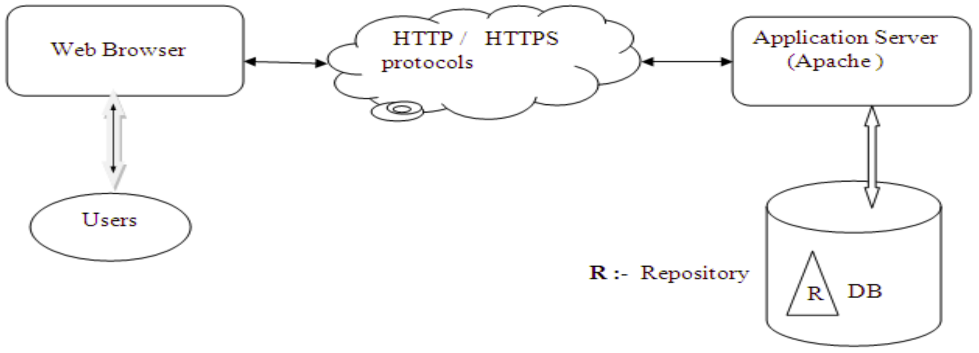
The user can perform various operations on the repository files. He can modify the existing files, delete files, add files, and view the log information. The changes are updated to the repository only after the user commits them. The user needs to perform an update operation to update the changes made by other users of the repository who shares it. Each time he commits a change made, it appears with a new version number or revision number. The user can even view what changes are being made by other user to the same repositories file content by using the command Diff,provided the revision numbers.

2.2. Admin Module:

The admin has to login into the site first. Once the admin logins, he can perform various operations such as view the list of newly registered users, he can approve them by selecting them .He can view what all repositories are present in his server, create new repositories, provide working space for the users in the server, delete the repositories etc. When the user approves the new users, he has to provide them with user id and password ant the details are passed to the approved users making use of an email sent automatically as he approves them. The admin has to enter the user id and password details to the subversion by entering them from the terminal using the command mandatorily

3. SYSTEM ARCHITECTURE

Here we use a 3-Tier Architecture.



3.1 Presentation layer using Browser technology:

Decoding URLs: protocol/host/file
Host name converted to IP address (164.11.8.19)
Issue request to remote server using appropriate protocol using HTTP
Accept the returned HTML file

3.2 An **Application** layer using a web application server platform + application programs
Interacting with the server (accessing input and generating output)
Interpreting the requests according to business rules and past transactions from this client
Requesting the appropriate data from the Persistence layer
Computing derived data
Creating the HTML for the page

33 Persistence layer using a relational database or other data store technology
Interaction with the database using standard languages

SVND Database:

The etymology of "FSFS" is based on Subversion's use of the term "file system" for its repository storage system. FSFS stores its contents directly within the operating system's file system, rather than a structured system like Berkeley DB. Thus, it is a "[Subversion] File System atop the File System"

SVN File system:

One can view the Subversion file system as "two-dimensional". Two coordinates are used to unambiguously address file system items: Path (regular path of Unix-like OS file system) Revision Each revision in a Subversion file system has its own root, which is used to access contents at that revision. Files are stored as links to the most recent change; thus a Subversion repository is quite compact. The system consumes storage space proportional to the number of changes made, not to the number of revisions [8]..

The Subversion file system uses transactions to keep changes atomic. A transaction operates on a specified revision of the file system, not necessarily the latest. The transaction has its own root, on which changes are made. It is then either committed and becomes the latest revision, or is aborted. The transaction is actually a long-lived file system object; a client does not need to commit or abort a transaction itself, rather it can also begin a transaction, exit, and then can re-open the transaction and continue using it [4] [5] [6]. Multiple clients can access the same transaction and work together on an atomic change, though no existing clients expose this capability.

4. AGILE Model (SDLC)

The stage of planning and development process involves defining, developing, testing, delivering, operating, and maintaining a software product. Different lifecycle models emphasize different aspects and no single lifecycle model is suitable for all software products[9]. A lifecycle model that is understood and accepted improves project communication and enhances project manageability, resource allocation, cost control, and product quality.

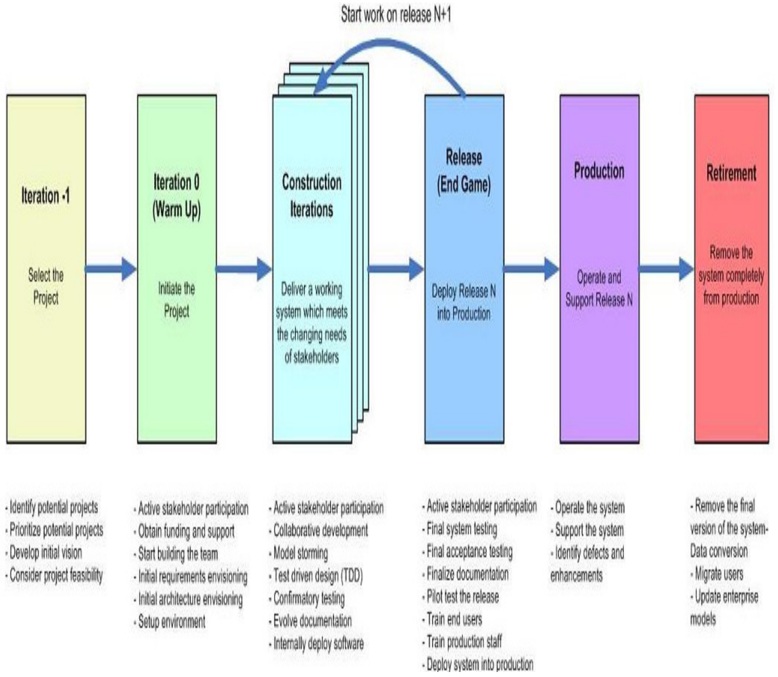

Start work on release N+1
Iteration -1
Select the Project
Iteration 0 (Warm Up)
Initiate the Project
Construction Iterations
Deliver a working system which meets the changing needs of stakeholders
Release (End Game)
Deploy Release N into Production
Production
Operate and Support Release N
Retirement
Remove the system completely from production
- Identify potential projects
- Prioritize potential projects
- Develop initial vision
- Consider project feasibility
- Active stakeholder participation
- Obtain funding and support
- Start building the team
- Initial requirements envisioning
- Initial architecture envisioning
- Setup environment
- Active stakeholder participation
- Collaborative development
- Model storming
- Test driven design (TDD)
- Confirmatory testing
- Evolve documentation
- Internally deploy software
- Active stakeholder participation
- Final system testing
- Final acceptance testing
- Finalize documentation
- Pilot test the release
- Train end users
- Train production staff
- Deploy system into production
- Operate the system
- Support the system
- Identify defects and enhancements
- Remove the final version of the system-Data conversion
- Migrate users
- Update enterprise models

Fig: Agile Life Cycle Model

5. IMPLEMENTATION

5.1 Introduction to PHP:

PHP is a widely-used open source general-purpose SERVER side scripting language that is especially suited for web development and can be embedded into HTML [10]

5.2 Introduction to HTML:

HTML stands for Hyper Text Markup Language. HTML is not a programming language, it is a markup language. A markup language is a set of markup tags. HTML uses markup tags to describe web pages. We have made use of HTML to design our front end part [10]..

5.3 Introduction to AJAX::

Asynchronous JavaScript and XML(AJAX) is the art of exchanging data with a server and update parts of a web page without reloading the whole page and is a technique for creating fast and dynamic web pages. It is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. With Ajax, web applications can send data to, and retrieve data from, a server asynchronously (in the background) without interfering with the display and behavior of the existing page.

6. CONCLUSION

In This project the usage of subversion in an easier and efficient way for the clients as it reduces the users work to download repository and also saves the users workspace. Each user need not maintain a working copy on their local systems. He can easily do it, in the server space allocated directly. Subversion Web Client provides advantages over existing systems like saves the local system memory. Direct accesses to the repository in the serverDoesn't take much time to checkout the repository if it's of large size.

7. FUTURE SCOPE

The present project has lack of few facilities like obtaining the deleted files and allotting limited workspace for the user in the server. Subversion Web Client to be made available in future release will contain those things and make it a User friendly Open Source.

Obtaining lock operation for a file, Revert operation, Removing lock for file. Creating new directories in a repository, Svn prop set command implementation, Svn status command implementation, Implementing svn move command

7. GLOSSARY

Term[8]	Description
Add	Used to add a file or directory to your working copy. The new items are added to the repository when it is committed.
Checkout	Downloading a copy of a SVN Repository to a location on the hard disk
Commit	Used to pass the changes in your local working copy back into the repository, creating a new repository revision.
Delete	Delete a versioned item (and commit the change) the item no longer exists in the repository after the committed revision.
Diff	Show Differences, Very useful when we want to see exactly what changes have been made
Head	most recent revision, the revision carrying the highest number
HEAD revision	The latest revision of a file or folder in the *repository*.
Log	The SVN Log contains a list of all the revisions along with

	the date, the author responsible for the commit and the log message. Also known as “History”.
History	Show the revision history of a file or folder. Also known as “Log”.
Repository	Central place where data is stored and maintained.
Revision	Every time you commit a set of changes, you create one new “revision” in repository. It is a set of changes that were made when that revision was created.
SVN	A frequently sedabbreviation for Subversion.
Update	This Subversion command pulls down the latest changes from the repository into your working copy, merging any changes made by others with local changes in the working copy.
Working Copy	This is your local “sandbox”, the area where you work on the versioned files, and it normally resides on your local hard disk. You create a working copy by doing a “Checkout” from a repository, and you feed your changes back into the repository using “Commit”.

REFERENCES

[1] Defining Subversion" by Spjut, R.J.(1979). British Journal of Law and Society 6 (2): 254- 261.

[2] Bezmenov (Ex-KGB), Yuri. ""Soviet subversion of Western Society (1983)". Yuri Bezmenov

[3] http://www.cyberciti.biz/tips/linux-unix-subversion-svn-howto.html] by Vivek Gite

[4] [http://www.howtoforge.com/apache-subversion-repository/]

[5] [https://help.ubuntu.com/community/Subversion]

[6] [https://help.ubuntu.com/community/AddUsersHowto]

[7] [http://svnbook.red-bean.com/en/1.2/index.html]

[9] [http://wiki.greenstone.org/wiki/index.php/Useful_SVN_Commands]

[10] [http://en.wikipedia.org/wiki/Agile_Modeling]

[11] php.net/manual/en/index.php

[12] http://en.wikipedia.org/wiki/TortoiseSVN

[13] www.stackoverflow.com

[14] http://en.wikipedia.org/wiki/TortoiseSVN

[15] www.subversion.apache.org/

[16] www.subversion.tigris.org/

[17] www.svnbook.red-bean.com/

[18] www.w3schools.com

About the Author

Mr Roshan Kavuri has obtained his B.E Degree from Andhra university during1988-92, and M.Tech (CSE) from JNT University, Kukatpally Hyderabad, He is having nearly 20 years experience in industry as well as a faculty of Computer Science and Information Technology departments. He is pursuing his PhD from JNTU Hyderabad, His areas of research includes Computer architecture, parallel computing, Operating Systems and Computer networks. Presently he is working as associate professor in JB.Institute of Engineering Technology Since 2004.