MASARYK UNIVERSITY
FACULTY OF INFORMATICS

Real-time collaboration in Komodo

MASTER'S THESIS

Bc. Matúš Makový

Brno, Spring 2015

Declaration

Hereby I declare, that this paper is my original authorial work, which I have worked out by my own. All sources, references and literature used or excerpted during elaboration of this work are properly cited and listed in complete reference to the due source.

Bc. Matúš Makový

Advisor: RNDr. Filip Nguyen

Acknowledgement

Abstract

Keywords

Contents

1 Introduction

Many software solutions enable people to create new things in a better and faster way. In most cases the resulting product should be so complex that one person is not enuogh for the successful and fast creation. Creators try to collaborate to achive a common goal. Among other opportunities and possibilities are collaboration and sharing the greatest benefits of the Internet.

At the beginning, as the Internet didn't have such capacity, people tried to use it just for sharing their drafts of work and sending them to each other. This enabled creators to copperate, but they could not work at the same time without having to synchronize their drafts. In other words, they had to find differences between their drafts and reflect them to each other's version. Information technologies had solution for this called Revision control. It had also many limitations, for example 2 people could not edit the same file in a project without having to resolve confilicts manualy when they tried to merge their work with collaborator's version

With the development of the Internet came a reasonable solution called real-time collaboration. Using this principle, author can see what his collaborator is doing in real-time and the manual synchronization or manual confilict resolution is not necessary. Information technologies take care of this synchronization and conflict resolution for the users.

In the theoretical part of this thesis, author deals with principles of real-time collaboration and describes some of the techniques used to implement real-time collaboration in software over the Internet. There is also a comparison of these techniques from various aspects. The practical part of this thesis deals with JBoss Data Virtualization and Komodo software as a new version of Teiid Designer developed by Red Hat that should use real-time collaboration in its upcomming release. Author recommends best technique for this authoring software regarding the requirements of data models and finds a suitable implementation in Java programming language.

2 Real-time collaboration

This chapter covers the basic overview of collaboration in general, description of real-time collaboration and description of difficulties with its implementation over the Internet. Last sections of this chapter describe three techniques used for implementation over the Internet and its properties.

2.1 Basic overview

Collaboration, as defined by English dictionary, is an act of working with another or others on a joint project. We can identify 2 types of collaboration over the Internet, non real-time collaboration and real-time collaboration.

In non real-time collaboration users work on separate copies of a project and then need to merge their changes into one final project. This type of collaboration needs to have a common shared repository in which both users commit their changes. It doesn't offer such flexibility as real-time collaboraion. Users have to check with each other on what part of project are they working, because the confilict resolution in such systems is not ideal and confilict has to be resolved manually. Examples of non real-time collaboration could be Revision control (Git, SVN), (TO DO) .

When using real-time collaboration, software creates an illusion that users are working on one shared copy of a document online. There is no requirement to commit changes to some kind of repository. Changes are reflected and saved immediately. Examples of real-time collaborative editors are Google Docs, Etherpad and Google Wave.

Software engineer has different options for implementetation of real-time collaboration in software solution.

Requirements for a good technique are:

- speed
- latency tolerantion
- low data transfer
- consistency maintanance
- good conflict resolution

The speed requirement means that changes made on one side of the collaboration process need to be reflected to the other sides as soon as possible and vice versa. If a collaboration technique is fast enough it is much easier to satisfy other requirements on this technique. Following two requirements are releated very closely to the speed requirement.

One of the problems of real-time collabarative editing could be the latency of the network, here comes the latency toleration requirement. Implemented technique should be able to tollarate the latency of the Internet, because collaborators could be in very distinct parts of the world. It should be able to reconstruct the right order of operations because packets sent over the Internet don't necessary come in right order.

Low data transfer requirement is also very important. Different sides of the collaboration should send as few data as possible. Transfered data should only describe that change that is needed to be done, it is not necessary to transfer the whole project. The less data is needed to transfer the faster the protocol can be.

Consistency maintanance is necesssary for the success, it has to be ensured, that users on both sides are looking and the same version of a document regardless of number and coplexity of operations done on both sides of the collaboration. Lack of consistency could cause other problems and chaos in the document versions.

Because the real-time collaboration is asychnronious there rises the problem of concurency. This means, that changes can happen at the same time and in the same sections of project. Good conflict resolution requirement is present because of concurency. Implementation techniques have to be able to resolve a conflict when users are editing the same part of the project.