

ACE

a collaborative editor

Pflichtenheft

Berner Fachhochschule
Hochschule für Technik und Informatik

Ausgabe vom:	05.04.2005
Version:	1.0
Projektteam:	Mark Bigler (biglm2@hta-bi.bfh.ch) Simon Räss (rasss@hta-bi.bfh.ch) Lukas Zbinden (zbinl@hta-bi.bfh.ch)
Empfänger:	Jean-Paul Dubois (doj@hta-bi.bfh.ch) Claude Fuhrer (frc@hta-bi.bfh.ch)
Ablage:	Subversion Repository



Inhaltsverzeichnis

1	Einleitung	4
1.1	Zweck des Dokumentes	4
2	Ausgangslage	4
3	Ist-Zustand	4
4	Ziele	5
4.1	Aufbau von Knowhow	5
4.2	Evaluation bestehender Algorithmen	5
4.3	Implementation Algorithmus	5
4.4	Testframework für Algorithmus	5
4.5	Konzept GUI	5
4.6	Konzept Netzwerk/Kommunikation	6
5	Nicht-Ziele	6
5.1	Ausarbeiten von Sicherheitsaspekten	6
5.2	Prototyp kollaborativer Texteditor	6
6	Anforderungen	6
6.1	Architektur	6
6.2	Technologie	6
6.3	Schnittstellen	6

Tabellenverzeichnis

1	Versionskontrolle	3
2	Prüfung/Genehmigung	3
3	Projektziele	5
4	Nicht-Ziele	6

Versionskontrolle

Version	Datum	Verantwortlich	Bemerkungen
0.1	15.03.2005	zbinl	Erste Version
0.2	16.03.2005	Projektteam	Überarbeitung
0.3	04.04.2005	Projektteam	Letzte Anpassungen
1.0	05.04.2005	Projektteam	Freigabe

Tabelle 1: Versionskontrolle

	Stelle	Datum	Visum	Bemerkungen
Freigegeben	Projektteam	05.04.2005		
Genehmigt				

Tabelle 2: Prüfung/Genehmigung

1 Einleitung

Im Projekt *ACE* soll ein kollaborativer, plattformunabhängiger Editor entwickelt werden. Diese Applikation ermöglicht mehreren Personen, ein Textdokument gemeinsam zu bearbeiten. Dabei arbeitet jede Person mit dem Editor an einem eigenen Computer. Alle Teilnehmer sind über ein Netzwerk verbunden und sehen jederzeit den gleichen Dokumentinhalt. Wenn jemand der Gruppe eine Änderung im Dokument vornimmt, wird dies in Echtzeit und synchron allen anderen Benutzern angezeigt. Jeder Benutzer hat dadurch den Überblick über alle Änderungen im Dokument. Dieser Editor ermöglicht zum Beispiel ein gemeinsames Brainstorming von mehreren Personen, welche sich an verschiedenen Orten befinden.

1.1 Zweck des Dokumentes

Das vorliegende Dokument beschreibt die Ziele, welche mit der angestrebten Lösung zu erreichen sind sowie die Anforderungen an das in der Semesterarbeit zu erstellende System.

2 Ausgangslage

Das gemeinsame Entwerfen eines elektronischen Dokumentes (z.B. Textdatei), wobei jeder Beteiligte mit seinem eigenen Computer arbeitet, ist noch praktisch unbekannt. Diese Technologie birgt ein grosses Anwendungspotenzial. Der zu entwickelnde Editor *ACE* soll komplett neuartige und effiziente Editier-Möglichkeiten bieten und dadurch neue Wege der Zusammenarbeit eröffnen. Bis heute existiert keine marktreife, plattformunabhängige Applikation dieser Art.

3 Ist-Zustand

Die dem kollaborativen Editor zugrunde liegende Theorie, bekannt als *Operational Transformation*, entstammt aus dem Forschungsgebiet der "Computer Supported Cooperative Work - CSCW". Applikationen aus diesem Bereich werden als "Groupware" bezeichnet. Seit Mitte der 90er Jahren sind zahlreiche Forschungsarbeiten geschrieben worden. Die meisten davon beinhalten theoretische Überlegungen, so zum Beispiel mathematische Beschreibungen oder Beweise zu Synchronisationsalgorithmen. Viel Theoriewissen wurde erarbeitet. Dieses Know-How soll nun in die konkrete Implementation eines kollaborativen Editors einfließen um daraus eine hochentwickelte, konkurrenzfähige Applikation auf den Markt zu bringen.

4 Ziele

In der Semesterarbeit soll die Basis gelegt werden für die Implementation eines kollaborativen und plattformunabhängigen Texteditors im Rahmen der Diplomarbeit.

Beschreibung	Priorität
Aufbau von Knowhow	1
Evaluation bestehender Algorithmen	1
Implementation Algorithmus	1
Testframework für Algorithmus	1
Konzept GUI	2
Konzept Netzwerk/Kommunikation	2

Tabelle 3: Projektziele

4.1 Aufbau von Knowhow

Es geht darum, ein fundiertes Basiswissen im Bereich des CSCW aufzubauen. Das Ziel ist, eine Übersicht über den aktuellen Forschungsstand und über die wichtigsten Errungenschaften in diesem Gebiet zu gewinnen. Das angeeignete Know-How soll beim Evaluieren der bestehenden Synchronisationsalgorithmen zum tieferen Verständnis beitragen und ein sachgerechtes Beurteilen ermöglichen.

4.2 Evaluation bestehender Algorithmen

Die Forschung hat seit Anfang der 90er Jahre zahlreiche Synchronisationsalgorithmen hervorgebracht. Das Ziel ist eine Übersicht über die bestehenden Algorithmen zu erarbeiten. Jeder evaluierte Algorithmus soll prinzipiell verstanden werden. Die Übersicht soll Vor- und Nachteile aufzeigen und es ermöglichen, den am besten geeigneten Algorithmus für einen kollaborativen Texteditor zu bestimmen.

4.3 Implementation Algorithmus

Nach Evaluation der Algorithmen soll einer oder eventuell mehrere implementiert werden. Die Implementation des Algorithmus wird mit dem Testframework geprüft.

4.4 Testframework für Algorithmus

Parallel zur Entwicklung des Synchronisationsalgorithmus soll ein Testframework erstellt werden. Dieses ermöglicht das sorgfältige Austesten des Algorithmus mit verschiedenen, klar definierbaren Abläufen.

4.5 Konzept GUI

In erster Linie soll eine Evaluation von Textkomponenten in Java erfolgen. Das Ziel ist herauszufinden, welche sich am besten für einen kollaborativen Texteditor eignen. Es muss möglich sein, verschiedene Benutzer respektive deren Aktionen in einer Textkomponente darzustellen.

4.6 Konzept Netzwerk/Kommunikation

Zwei Themen müssen erarbeitet werden: Das dynamische Auffinden aller vorhandenen Instanzen des Editors in einem Netzwerk sowie der Nachrichtenaustausch zwischen den Instanzen. Es sollen die in diesem Themengebiet vorhandenen Technologien studiert werden. Daraus entsteht ein für einen kollaborativen Texteditor angepasstes Konzept.

5 Nicht-Ziele

Beschreibung
Ausarbeiten von Sicherheitsaspekten Prototyp kollaborativer Texteditor

Tabelle 4: Nicht-Ziele

5.1 Ausarbeiten von Sicherheitsaspekten

Es ist nicht das Ziel Sicherheitsaspekte wie Authentifikation, Authorisierung und Verschlüsselung zu berücksichtigen. Es wird auch kein Konzept dafür erstellt.

5.2 Prototyp kollaborativer Texteditor

Im Rahmen der Semesterarbeit soll kein Prototyp eines kollaborativen Editors entstehen.

6 Anforderungen

6.1 Architektur

Die Architektur wird basierend auf dem Peer-to-Peer Modell entwickelt. Die Applikation soll dadurch ohne einen zentralen Server funktionieren.

6.2 Technologie

Der zu entwickelnde kollaborative Texteditor soll plattformunabhängig sein. Darum erfolgt die Implementation mit der Programmiersprache *Java*.

6.3 Schnittstellen

Durch die Offenlegung des internen Kommunikationsprotokolls von *ACE* haben andere Applikationsentwickler die Möglichkeit, direkt mit dem Editor zu kommunizieren. Dadurch entstehen beliebige Erweiterungsmöglichkeiten mit dem kollaborativen Texteditor.