

**Struktur und Werkzeuge des
experiment-spezifischen Datenbereichs
der
SFB501 Erfahrungsdatenbank**

Stefan Vorwieger, Holger Damczyk, Marion Fechtig, Boris Schadt, Oliver Swienty
{ vorwiege, damczyk, fechtig, schadt, swienty }@informatik.uni-kl.de

11/1999

Sonderforschungsbereich 501

Arbeitsgruppe Software Engineering
Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
67653 Kaiserslautern
Germany

Zusammenfassung

Software-Entwicklungsartefakte müssen zielgerichtet während der Durchführung eines Software-Projekts erfaßt werden, um für die Wiederverwendung aufbereitet werden zu können. Die methodische Basis hierzu bildet im Sonderforschungsbereich 501 das Konzept der Erfahrungsdatenbank. In ihrem experiment-spezifischen¹ Datenbereich werden für jedes Entwicklungsprojekt alle Software-Entwicklungsartefakte abgelegt, die während des Lebenszyklus eines Projektes anfallen. In ihrem übergreifenden Datenbereich werden all diejenigen Artefakte aus dem experiment-spezifischen Datenbereich zusammengefaßt, die für eine Wiederverwendung in nachfolgenden Projekten in Frage kommen.

Es hat sich gezeigt, daß bereits zur Nutzung der Datenmengen im experiment-spezifischen Datenbereich der Erfahrungsdatenbank ein systematischer Zugriff notwendig ist. Ein systematischer Zugriff setzt jedoch eine normierte Struktur voraus.

Im experiment-spezifischen Bereich werden zwei Arten von Experimenttypen unterschieden: 'Kontrollierte Experimente' und 'Fallstudien'. Dieser Bericht beschreibt die Ablage- und Zugriffsstruktur für den Experimenttyp 'Fallstudien'. Die Struktur wurde aufgrund der Erfahrungen in ersten Fallstudien entwickelt und evaluiert.

1. Im SFB 501 werden alle Software-Entwicklungen im Sinne eines Software-Engineering-Experiments (d.h. Fallstudie oder kontrolliertes Experiment) durchgeführt.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Kontext	1
1.2	Erfahrungsdatenbank	1
1.3	Datenbereich "Fallstudien"	2
1.4	Zugriffs- und Ablagestruktur	3
1.5	Evaluierung	5
1.6	Intention des Berichts	6
2	Zugriffstruktur für Team-Projekte (CoDEX-Typ)	7
2.1	Team-Projekte	7
2.2	Strukturbeschreibung	7
2.3	Technische Vorbereitung	8
2.4	Charakterisierung (QIP1)	8
2.5	Ziele (QIP2)	9
2.6	Prozeßwahl/Projektplan (QIP 3)	10
2.7	Durchführung (QIP4)	12
2.8	Analyse (QIP5)	12
3	Zugriffstruktur für "technische" Praktika (SE-I-Typ)	13
3.1	"Technische" Praktika	13
3.2	Strukturbeschreibung	13
3.3	Technische Vorbereitung	13
3.4	Charakterisierung (QIP1)	14
3.5	Ziele(QIP2)	14
3.6	Prozeßwahl/Projektplan (QIP3)	16
3.7	Durchführung (QIP4)	18
3.8	Analyse (QIP5)	18
4	Zugriffstruktur für "technische & organisatorische" Praktika (SE-II-Typ)	19
4.1	"Technische & organisatorische" Praktika	19
4.2	Strukturbeschreibung	19
4.3	Technische Vorbereitung	19
4.4	Charakterisierung (QIP1)	20
4.5	Ziele (QIP2)	20
4.6	Prozeßwahl/Projektplan (QIP3)	22
4.7	Durchführung (QIP4)	23
4.7.1	Subprojekt, SE-II-Typ	24
4.8	Analyse (QIP5)	26
5	Gegenüberstellung der Zugriffsstrukturen für Fallstudien	27
5.1	Allgemein	27
5.2	Technische Vorbereitung	28
5.3	Charakterisierung (QIP1)	28
5.4	Ziele (QIP2)	29
5.5	Prozeßwahl/Projektplan (QIP3)	30
5.6	Durchführung (QIP4)	32
5.7	Analyse (QIP5)	34
6	Ablagestruktur für Fallstudien	35

6.1 Allgemein	35
6.2 Inhalte	36
6.2.1 CHARAKTERISIERUNG	36
6.2.2 DURCHFUEHRUNG	36
6.2.3 MODELLE	36
6.2.4 PRODUKTE	37
6.2.5 VORBEREITUNG	38
6.2.6 ZIELE	38
6.3 Gesamtstruktur	39
7 Automatisierung	41
7.1 Problembesprechung	41
7.2 Überblick über die praktische Umsetzung des Eingabevorgangs	42
7.3 Struktur des Eingabedokuments	43
7.3.1 Hauptabschnitt A: Inhaltliche Gesamtübersicht	43
7.3.2 Hauptabschnitt B: Festlegung allgemeiner Eingabeparameter	44
7.3.3 Hauptabschnitt C: Festlegung des Kontext-Vektors	48
7.3.4 Hauptabschnitt D: Festlegung der Eingabeparameter bez. der Team-Mitglieder des Experiments	49
7.3.5 Hauptabschnitt E: Erzeugen und Einfügen der Dokumente in die EDB	52
7.3.6 Hauptabschnitt F: Hilfeabschnitt bez. der Eingabe deutscher Umlaute und des "ß"	53
8 Zusammenfassung und Ausblick	55
8.1 Zusammenfassung	55
8.2 Ausblick	55
8.2.1 Sichtenbasierte Zugriffe	55
8.2.2 Schnittstelle für die Einlagerung	55
8.2.3 Evolution der Struktur	56
8.2.4 Generierung	56
8.3 Danksagung	57
Anhang A Literatur	A-1
Anhang B Abkürzungen der Ablagestruktur	A-3
Anhang C Vorgehen bei Änderungen des Generierungs-Systems	A-5
C.1 Hinzufügen eines neuen Experimenttyps	A-5
C.1.1 Modifikationen im Dokument "spezifisch_de.html"	A-5
C.1.2 Modifikationen im Eingabedokument	A-6
C.1.3 Modifikationen im CGI-Skript	A-6
C.2 Hinzufügen eines neuen Attributs des Kontextvektors	A-6
C.2.1 Modifikation des Eingabedokuments	A-6
C.2.2 Modifikation des CGI-Skriptes	A-7
C.3 Ändern der Experimentverzeichnisstruktur	A-8
C.3.1 Modifikation des CGI-Skripts	A-8
Anhang D Dokumentation des Generierungs-Systems	A-11
D.1 Skript-Erläuterung	A-11
D.2 Skript-Beschreibung	A-11
D.3 Das Skript aufrufende Files	A-12
D.4 Vom Skript modifizierte Files	A-12
D.5 Vom Skript erzeugte Files	A-12
D.6 Aufbau des Skriptes	A-13
D.7 Obligatorische und optionale Skript-Parameter	A-14

1 Einleitung

1.1 Kontext

Ziel des Sonderforschungsbereichs 501 (SFB 501) ist es, die Entwicklung großer, komplexer Softwaresysteme durch Einsatz von Software-Engineering-Erfahrung (SE-Erfahrung) und generischen Methoden¹ zu verbessern [1]. SE-Erfahrungen sind dabei alle Artefakte, die

- in nachfolgenden Entwicklungsprojekten wiederverwendet werden können: (z.B. Quell-Code oder Systementwürfe, Qualitätsmodelle), oder
- zur Wiederverwendung dieser Artefakte beitragen können: (z.B. Meßdaten zur Projektkontrolle und Post-Mortem-Analyse, Planungsartefakte wie den Projekt- oder Meßplan sowie jegliche Art von qualitativer Erfahrung, z.B. in Form von 'Lessons learned').

SE-Erfahrungen müssen schritthaltend und systematisch während der Planung und Durchführung von Software-Entwicklungsprojekten erfaßt werden. Fokus der Erfassung ist zum einen

- die Verbesserung der Projektkontrolle während der Durchführung durch ständigen Abgleich mit den Soll-Daten der Planung, und zum anderen
- die Verfügbarkeit von Projektdaten, die eine Analyse nach dem Projekt (Post-Mortem) und die Aufbereitung deren Ergebnisse für die Wiederverwendung in nachfolgenden Projekten ermöglicht.

Software-Entwicklungsprojekte, bei denen zielgerichtet und projektbegleitend SE-Erfahrungen erfaßt werden, werden innerhalb des SFB grundsätzlich als Software-Engineering-Experimente verstanden, d.h. sie werden entweder als kontrollierte SE-Experimente [2] oder als Fallstudien [3] durchgeführt. Die Erfassung geschieht im Umfeld einer 'Experience Factory' [4], die den organisatorischen Rahmen für die Wiederverwendung bereitstellt. Zentrale technische Komponente ist dabei die *Erfahrungsdatenbank*, in der alle SE-Erfahrungen eines Experiments sowie alle Informationen, die durch Generalisierung aus einem Experiment gewonnen wurden, schließlich abgelegt und zugreifbar sind.

1.2 Erfahrungsdatenbank

Die Erfahrungsdatenbank des SFB (SFB 501-EDB) unterscheidet für den Zugriff und die Ablage von SE-Erfahrungen zwei Bereiche: Ein experiment-spezifischer und ein experiment-übergreifender Bereich (s. Abb. 1). Im experiment-spezifischen Bereich werden primär SE-Erfahrungen der einzelnen Experimenten strukturiert abgelegt und zugreifbar gemacht. Für jedes neue Experiment wird ein neuer Datenbereich zur Verfügung gestellt. Für den experiment-spezifischen Bereich werden —gemäß der zwei

1. Unter dem Begriff „generische Methoden“ werden im SFB alle Beschreibungs- und Generierungstechniken mit den dazugehörigen Werkzeugen zusammengefaßt, durch die eine Wiederverwendung von existierenden Softwareprodukten, Entwicklungsschritten und Erfahrungen bei der Entwicklung eines neuen Systems methodisch unterstützt wird.

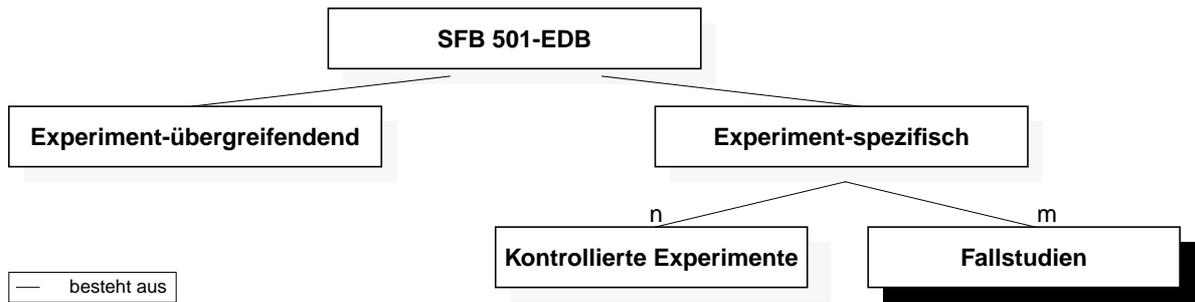


Abb. 1: Die Bereiche der SFB 501-EDB

Typen von Experimenten— Fallstudien und kontrollierte Experimente unterschieden (s. Abb. 1). Fallstudien sind im wesentlichen Software-Entwicklungsprojekte, die neben der Erstellung eines Software-Artefakts auch die zielgerichtete und begleitende Erfassung von Meßdaten zur Bewertung von Produkten und Prozessen zum Ziel hat. Kontrollierte Experimente dagegen fokussieren meist einem Ausschnitt des Entwicklungsprozesses und stellen zur Durchführung kontrolliert die gewünschte Umgebung her. Auf diese Weise sind genauere Bewertungen möglich, jedoch sind diese nur für diese spezielle Umgebung gültig.

Im experiment-übergreifenden Bereich werden SE-Erfahrungen hauptsächlich im Hinblick auf Wiederverwendung und zielgerichteter Verbesserung über Experimente hinweg abgelegt. Er ist daher über eine Reihe von Experimenten gültig und wird jeweils durch neue Experiment-Daten angereichert. Die Evolution dieses Datenbereichs wird gegebenenfalls durch eine interne Versionierung dokumentiert.

Im weiteren wird in diesem Bericht auf den Bereich "Fallstudien" des experiment-spezifischen Bereichs eingegangen. Für diesen Bereich lagen schon zu Beginn des Aufbaus der Erfahrungsdatenbank eine Vielzahl von Fallstudien vor, so daß eine Festigung der anfänglichen Datenstrukturen sowie Evaluierung anhand von neueren Fallstudien möglich waren.

1.3 Datenbereich "Fallstudien"

Im Datenbereich "Fallstudien" wurde eine weitere Verfeinerung in unterschiedliche Fallstudien-Typen vorgenommen. Zum einen dient diese Klassifizierung der weiteren Übersicht über die immer weiter wachsende Anzahl von Fallstudien. Zum anderen ist auch in Hinblick auf unterschiedliche Strukturen des Zugriffs der einzelnen Typen eine weitere Unterscheidung sinnvoll und für die Generierung von Strukturen sogar absolut notwendig.

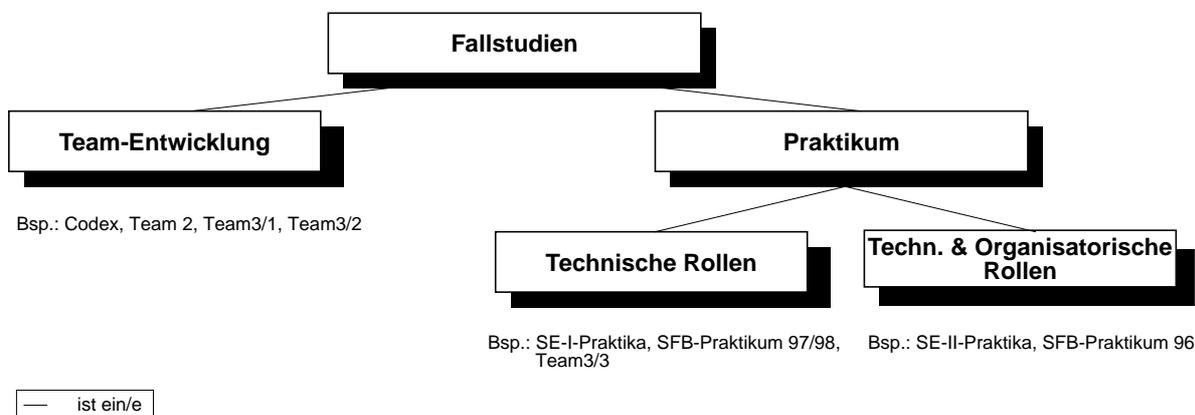


Abb. 2: Der Bereich "Fallstudien" der SFB 501-EDB

Eine erste Unterscheidung wird zwischen Experimenten, die als Studenten-Praktikum, und Experimenten, die in SFB-Teams durchgeführt wurden, getroffen (s. Abb. 2). Ein Studenten-Praktikum bedarf einer Vorbereitung über Aushänge, in denen um Teilnehmer und unterstützende Hilfskräfte geworben wird. Während der Durchführung eines Praktikums findet eine regelmäßige Kontrolle der Betreuenden über Aufgabenstellungen statt, welche in Übungsblättern definiert und in den entsprechenden Sitzungen überprüft werden. Für Team-Experimente dagegen werden Teilnehmer extern bestimmt bzw. zur Verfügung gestellt. Bei der Durchführung werden ebenfalls Milestones überprüft, jedoch sind diese nicht an Aufgabenstellungen, sondern am Entwicklungsprozeß orientiert.

Im Großen und Ganzen sind die beschriebenen Unterschiede zwischen Praktika und Team-Experimenten zunächst gering. Jedoch existiert neben dem Typ "technisches Praktikum", der einer Team-Entwicklung ähnelt, ein weiterer Praktikumstyp "technisch & organisatorisch". Hierbei erhalten die Praktikumsmitglieder die Aufgabe, selbst ein Entwicklungsexperiment zu planen (organisatorische Rollen), es durchzuführen (technische Rollen) und dessen Durchführung zu kontrollieren und zu verwalten (wieder organisatorische Rollen). In den experiment-spezifischen Datenbereich muß also ein Experiment im Experiment eingelagert werden. Hierzu wird entsprechend die Durchführung des Experiments als ein eigenes Sub-Experiment betrachtet, so daß eine sinnvolle Ablage und Zugriff möglich ist.

1.4 Zugriffs- und Ablagestruktur

Jedes Software-Engineering Experiment im SFB 501 wird systematisch auf der Basis des Quality Improvement-Paradigm [5] durchgeführt. Es beschreibt die Planung, Durchführung und Aufbereitung der Experimentergebnisse in sechs Schritten: die Planung eines Experiments wird in drei Schritte verfeinert (Charakterisierung des Experiments, Zielsetzung und Erstellen eines Projektplans); ein Schritt beschreibt die eigentliche Durchführung des Experiments; die Aufbereitung der Projektergebnisse zu wiederverwendbaren, generalisierten Ergebnissen findet in Schritt fünf und sechs statt (Analyse der Meßdaten, und qualitativen Erfahrungen und Ableitung von generalisierten Modellen sowie die Integration der Ergebnisse in den übergreifenden Datenbereich).

Abb. 3 zeigt die Einbettung beider Bereiche der SFB 501-Erfahrungsdatenbank in den QIP-Zyklus. In den ersten drei Planungsschritten wird im wesentlichen lesend auf den übergreifenden Bereich zugegriffen. Die dort entnommenen Informationen werden an die Charakteristika und Ziele des aktuellen Experiments angepaßt und in den speziell für dieses Experiment angelegten spezifischen Bereich eingelagert. In diesem Kontext haben wir die Planung um einen Schritt "Organisatorische Vorbereitung" erweitert. In ihm werden die Ergebnisse aller organisatorischen Tätigkeiten im Vorfeld der eigentlichen technischen Planung eingeordnet: z.B. Mitteilungen, die Entscheidungen für die Planung enthalten oder Aushänge und Newsgruppen-Artikel für die Anwerbung von Entwicklern und anderen Hilfskräften. Diese Ergebnisse sind ebenfalls Teil des experimentspezifischen Datenbereichs. Ebenso werden alle Entwicklungser-

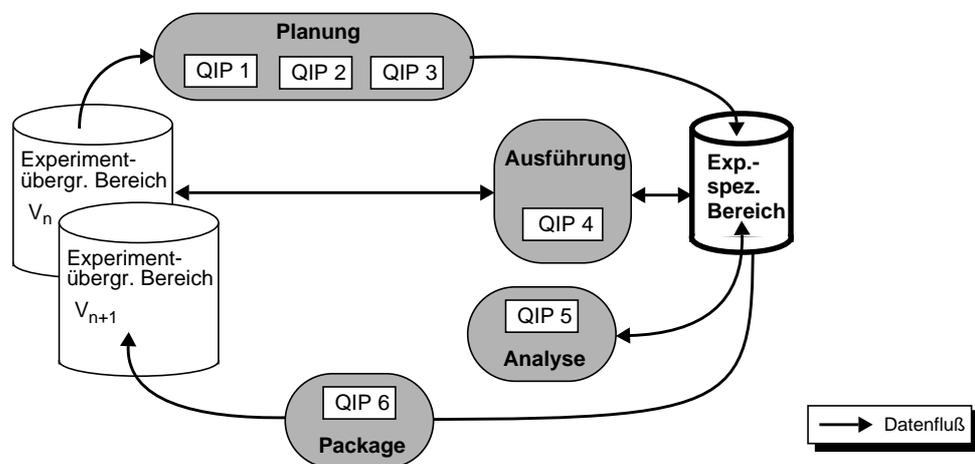


Abb. 3: Die Rolle der SFB 501-EDB im Lernzyklus QIP

gebnisse, Meßdaten sowie alle qualitativen Erfahrungen (z.B. lessons learned) des QIP-Schrittes vier im spezifischen Bereich abgelegt. Im ersten Analyse-Schritt (QIP 5) findet die Aufbereitung der Rohdaten von Messungen und qualitativer Erfahrung statt. Diese Daten sind immer noch eng mit dem Experiment verbunden, weshalb sie im experiment-spezifischen Bereich abgelegt werden. Generalisierte Ergebnisse der Analyse wie z.B. Modelle werden dagegen im experiment-übergreifenden Bereich der Datenbank abgelegt, da sie allgemeine, für eine Reihe von Experimenten gültige Ergebnisse repräsentieren. Diese fallen hauptsächlich in Schritt sechs des QIP-Zyklus an und sind nicht Teil des experiment-spezifischen Bereichs. Zusammengefaßt enthält der experiment-spezifische Datenbereich alle SE-Erfahrung der Planung und Ausführung sowie alle SE-Erfahrungen in der Analyse, die für das Experiment und dessen spezifischen Kontext gültig sind.

Für den Zugriff auf die eingelagerten SE-Erfahrungen werden zwei Möglichkeiten angeboten (s. Abb. 4): zum einen wird eine spezielle Zugriffsebene zur Verfügung gestellt, welche über eine Reihe von mit Hyper-Links verbundenen HTML-Seiten realisiert ist. Zum anderen ist auch mit dem HTML-Browser eine Navigation auf dem Filesystem direkt möglich (Direktzugriff). Beide Zugriffsarten [6] besitzen unterschiedliche Eigenschaften.

Ein Zugriff der SE-Erfahrungen im experiment-spezifischen Datenbereich über die HTML-Zugriffsstruktur ist anhand der QIP-Schritte organisiert. Da die einzelnen QIP-Schritte den entsprechenden Phasen eines Experiments entsprechen, ist anhand des Stands der einzelnen Unterbereiche der Zugriffsstruktur der Stand des Projekts ablesbar. Ebenso leicht fällt auch die Suche nach bestimmten SE-Erfahrungen, wenn bekannt ist, in welcher Phase des Experiments das entsprechende Artefakt erstellt werden sollte. Des weiteren stehen zur Durchführung eines QIP-Schrittes alle vorhandenen Dokumente des vorangegangenen Schritts direkt zur Verfügung, d.h. durch die Strukturierung nach dem QIP wird die Durchführung des Experiments weiter unterstützt. Nachteil dieser Art des Zugriffs ist zum einen, daß Wissen über ein Experimentablauf nach dem QIP voraussetzt wird. Andererseits weisen die einzelnen Fallstudien unterschiedliche Ausprägungen auf den unteren (verfeinerten) Strukturebenen auf, z.B. für die beiden Arten von SE-Praktika: Die Variante "technisch & organisatorisch" enthält einen weiteren Sub-QIP-Zyklus der Schritte zwei bis fünf. Der Grund hierfür ist, daß die zentrale Aufgabe des Praktikums darin besteht, ein eigenes Experiment zu planen und durchzuführen.

Die zweite Art des Zugriffs auf experiment-spezifische SE-Erfahrungen ist der Direktzugriff auf die Ablagestruktur. SE-Erfahrungen sind hier nach Artefakt-Typen strukturiert. Die Struktur wird ausschließlich durch einen Baum von Verzeichnissen auf dem Dateisystem (Unix) gebildet. Die Ablage-

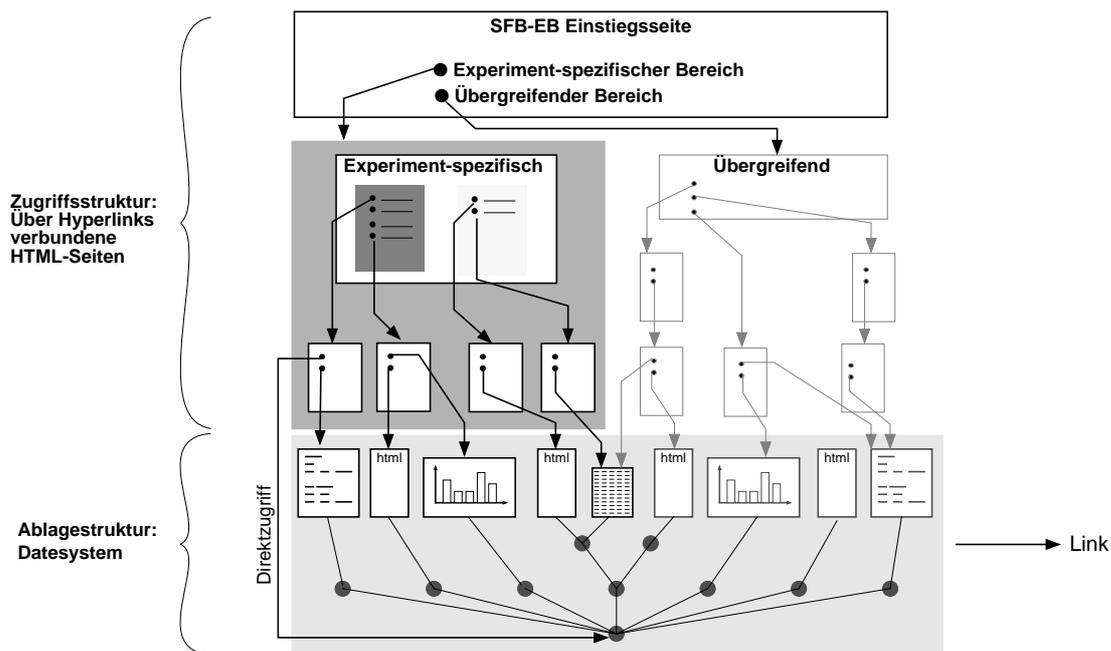


Abb. 4: Die Ablage und Zugriffsstruktur der Erfahrungsdatenbank

struktur hat den Vorteil, daß sie für alle Typen von Fallstudien und für die Experimente gleich ist. Desweiteren setzt sie kaum Wissen über ein Lebenszyklusmodell wie das QIP voraus und unterstützt die Suche nach einem bestimmten Artefakt-Typ auf direktem Weg. Andererseits läßt sich über sie nur schwer beurteilen, in welchem Stadium das Experiment sich zum Zugriffszeitpunkt befindet.

Beide Zugriffsarten stellen zwei feste Sichten auf die SE-Erfahrungen eines Experiments dar, welche primär die Zugriffsart "Browsing" unterstützen. Andere Arten des Suchens, wie z.B. nach Inhalten von SE-Erfahrungen, sind nur in beschränktem Maße mittels besondere Befehle (z.B. *grep* oder *find*) auf dem Dateisystem möglich.

Beide Sichten müssen ständig konsistent gehalten werden. Unterstützt wird die Wartung durch:

- ein Generierungssystem, welches zu Beginn eines Experiments die vollständige Zugriffs- und Ablagestruktur erzeugt und mit initialen Informationen, wie der Name des Experiments oder der Teammitglieder, füllt,
- eine feste, dokumentierte Beziehung zwischen Ablage- und Zugriffsstruktur. (Hierin ist bspw. beschrieben, daß im Ablagesystem Prozeß- und Qualitätsmodelle im Verzeichnis "Modelle" zu finden sind und diese in der HTML-Zugriffsstruktur unter QIP-Schritt 3, Unterpunkt "Projektplan", zu finden sein müssen), sowie
- die dokumentierte Systematik beider Strukturen (QIP und Artefakt-Typ).

In den folgenden Kapiteln wird der Datenbereich "Fallstudien" näher erläutert. Hierzu wird in Kapitel 2 die Struktur der HTML-Zugriffsart besprochen. Hierzu wird zunächst in einer Gegenüberstellung auf Gemeinsamkeiten und Unterschiede der verschiedenen HTML-Strukturen für Team-Projekte und den unterschiedlichen Praktika eingegangen. Das Kapitel schließt mit der ausführlichen Darstellung jeder einzelnen Zugriffsstruktur. In Kapitel 3 wird auf Inhalt und Struktur des Ablagesystems auf Dateisystem-Ebene eingegangen. Kapitel 4 behandelt ausführlich die Generierung der initialen Ablage- und Zugriffsstruktur. Der Bericht schließt mit einer Zusammenfassung und der Diskussion möglicher Verbesserungen. Der Anhang beschreibt neben der Progammdokumentation des Generierungssystems, wie vorzugehen ist, wenn bestimmte Formen der Erweiterung am Generierungssystem vorgenommen werden sollen.

1.5 Evaluierung

Die Zugriffs- und Ablagestrukturen wurden anhand der folgenden Experimente erstellt und angepaßt [7].

Fallstudien (Team-Projekte)

Für den Fallstudientyp "Team-Projekte" wurden bisher fünf Fallstudien abgelegt. In *Team1*¹, welches 1995/96 durchgeführt wurde, wurden Anforderungsbeschreibungssprachen untersucht und Anforderungen in Statemate, SDT und SCR für ein Gebäudeautomationssystem erstellt und verglichen [8]. *Team2*² war eine dazu parallel verlaufende Fallstudie, in der auf der Basis einer reduzierten Anforderungsspezifikation für ein Gebäudeautomationssystem eine vollständige Entwicklung bis hin zu einem lauffähigen System stattfand [9]. *CoDEX*³ wurde im Anschluß an diese beiden Fallstudien aufgesetzt (1996/97). Es enthält eine ausführlichere Projektplanung und -kontrolle sowie genauere Zieldefinition für die Meßdatenerfassung. Auch hier wurde ein lauffähiges Gebäudeautomationssystem mit einer grafischen Benutzerschnittstelle erstellt [10]. Der hohe Grad an Vollständigkeit dieser Fallstudie war der Grund dafür, daß *CoDEX* als Basis für die Entwicklung der Zugriffsstruktur dieses Typs diente. Anhand der zeitlich darauf folgenden Fallstudien RTP⁴ [11], BaX⁵ [12] und BaX2⁶ wurde die Zugriff- und Ablagestruktur evaluiert und weiter verbessert und ergänzt.

1. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/team1_contents.html
2. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/team2_contents.html
3. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/codex_contents.html
4. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/rtp_contents.html
5. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/baX_contents.html
6. http://sep1.informatik.uni-kl.de:18000/EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN/baX2_contents.html

Fallstudien ("technische" Praktika)

Im Bereich der "technischen" Praktika wurden bisher vier Fallstudien abgelegt. Alle Praktika hatten zum Ziel, ein Steuerungssystem zu entwickeln. Praktikum *SE-I 97* wurde zunächst eingelagert und die Zugriffstruktur daran angepaßt. Nachträglich wurde dann das im Vorjahr durchgeführte Praktikum *SE-I 96* hinzugefügt. Anhand dieses und der in den beiden nächsten Jahren folgenden Praktika (*SE-I 98* und *SE-I 99* [13]) wurde die Zugriffstruktur bewertet und überarbeitet.

Fallstudien ("techn. & organ." Praktika)

Es wurden bisher ebenfalls vier "techn. & organ." Praktika im Fallstudien-Bereich der Erfahrungsdatenbank abgelegt. Grundlage für die Erst-Entwicklung der Zugriffstruktur war das Praktikum *SE-II 96/97*. *SE-II 95/96* wurde wiederum nachträglich eingelagert. Dies und die beiden folgenden Praktika (*SE-II 97/98* und *SE-II 98/99*) wurden zur Evaluierung der initialen Struktur eingelagert. Daraus resultierende Anpassungen führten zu der in den folgenden Kapiteln beschriebenen Struktur.

1.6 Intention des Berichts

Dieser Bericht soll einen Ausschnitt der Erfahrungsdatenbank des SFB, welche sich gerade im Aufbau befindet, dokumentieren und den aktuellen Stand des Wissens in diesem Bereich fixieren. Weiter soll er alle diejenigen unterstützen, die entweder in dem Bereich *Fallstudien* des experimentspezifischen Teils nach SE-Artefakten suchen oder solche einlagern wollen.

2 Zugriffstruktur für Team-Projekte (CoDEX-Typ)

2.1 Team-Projekte

Team-Projekte sind eine Form von Fallstudien, in denen Entwickler wie Planer und Manager hauptsächlich aus wissenschaftlichen Mitarbeitern gestellt werden. Fallstudien dieser Art ähneln am meisten "industriellen" Entwicklungsprojekten in Bezug auf die Organisationsform. Mitarbeiter stehen zu festen Arbeitszeiten zur Verfügung und besitzen bestimmte Vorkenntnisse bzw. die Fähigkeit, sich schnell an neue Anforderungen anzupassen. Diese Voraussetzungen sind in Studenten-Praktika nicht gegeben, weswegen Praktika in einem etwas anderen organisatorischen Umfeld ablaufen. Dies spiegelt sich auch in der Zugriffsstruktur wider.

Der Fallstudien-Typ "Team-Projekte" wird auch CoDEX-Typ genannt. CoDEX war das erste, weitgehend vollständige Team-Projekt, welches zum Aufbau der Zugriffs- und Ablagestruktur verwendet wurde.

2.2 Strukturbeschreibung

In den folgenden Kapiteln wird die HTML-Zugriffsstruktur mit Hilfe von Tabellen beschrieben. Die Spalte *Struktur* beschreibt fixe Strukturelemente, die in einem leeren Zugriffsschema bereits fest vorgegeben sind (d.h. generiert werden können bzw. worden sind). Inhalte der grau schraffierten Felder erscheinen zusätzlich in einer Inhaltszusammenfassung der HTML-Seite, die der Vereinfachung der Navigation dient. Unter der Spalte *Einträge* werden (Vorschläge für) Einträge beschrieben, die manuell eingetragen werden müssen/können, falls Inhalte dieser Art in der jeweiligen Fallstudie vorhanden sind. Diese Spalte erhebt nicht den Anspruch, vollständig zu sein.

Die Spalte *Ablage* enthält Verweise auf den Ablageort, d.h. auf ein Verzeichnis im Verzeichnisbaum des Ablagesystems. Das Verzeichnis wird durch ein Kürzel angegeben, welches bei der Beschreibung der Ablagestruktur (Kapitel 6) dokumentiert ist. Die Spalteninhalte beschreiben die Abbildung der Zugriffs- auf die Ablagestruktur.

Aufgrund der inhaltlichen Strukturierung (gem. QIP) kann es vorkommen, daß ein Eintrag für ein SE-Artefakt in unterschiedlichen Schritten erstellt werden kann, d.h. an unterschiedlichen Stellen in der Struktur sinnvoll wäre. In solchen Fällen haben wir uns darauf geeinigt, die SE-Erfahrung an einer ausgewählten Position in der Struktur wie gehabt einzutragen und alle weiteren Positionen mit Links (markiert mit → <Ziel>) auf andere Abschnitte einzutragen.

Im Anschluß an die tabellarische Darstellung der Struktur befinden sich zu den Strukturelementen kurze Erläuterungen, die sich nicht offensichtlich aus dem Namen des Strukturpunktes ergeben bzw. zu denen es in der Vergangenheit immer wieder Fragen von Benutzern gegeben hat.

Die Struktur stellt eine Obermenge über alle bisherigen Einträge der unterschiedlichen Fallstudien dar. Die einzelnen Fallstudien werden also immer Lücken aufweisen. Um zu demonstrieren, daß bewußt

Informationen für eine konkrete Fallstudie nicht vorhanden sind, wird zunächst eine vollständige Zugriffs- und Ablagestruktur generiert, die *nur* noch mit Inhalt gefüllt werden muß. "Freie Plätze" werden nicht mehr entfernt, sondern weisen dann auf Lücken hin.

2.3 Technische Vorbereitung

Hierunter werden alle Dokumente zusammengefaßt, die vor der eigentlichen Planung des Projekts erstellt werden müssen bzw. wiederverwendet werden können. Sie dienen der Vorbereitung der Projektplanung.

Technische Vorbereitung			
Struktur		Einträge	Ablage
Aushänge/Postings	Teammitglieder-Suche	WWW-Seiten, Aushänge, News-posting, Verpflichtung von HiWiS/AG-Mitarbeitern	V
	Externe Aufrufe (zur Nutzung)	email	V
	Erste Besprechungen	email, WWW-Seiten	V
Planungstemplates		Abstraction Sheets (leer)	PTEUT

Aushänge/Postings

Alle Dokumente zur Suche von Teammitgliedern bzw. Suche/Informierung von externen Projektinteressenten.

Planungstemplates

Wiederverwendbare Strukturdokumente zur Vorbereitung der Planung.

2.4 Charakterisierung (QIP1)

Dieser Abschnitt faßt alle SE-Erfahrungen zusammen, die im QIP-Schritt 1 erstellt werden sollten bzw. könnten.

Charakterisierung (QIP1)		
Struktur	Einträge	Ablage
Kontextbeschreibung	Kontextvektor	CV
Teammitglieder	Text-Liste, html-Seite	C
Projektankündigung	Aushang, WWW-Seiten	C
Beschreibung der einzusetzenden Technologien	→ QIP3	-

Projektankündigung

Dieser Strukturpunkt enthält den Ausschnitt der Projektankündigung, der das Projekt inhaltlich charakterisiert.

Beschreibung der einzusetzenden Technologien

Diese werden in QIP3 noch einmal aufgeführt und werden daher an dieser Stelle als Link eingetragen.

2.5 Ziele (QIP2)

Unter diesem Strukturpunkt werden alle Ziele der Fallstudie dokumentiert. Hierbei wird zwischen dem Projektanteil (*Projekt*) und dem Experimentanteil (*Lernen*) unterschieden. Der Projektanteil enthält alle Zielsetzungen, die die Entwicklung der geplanten Software sowie die Kontrolle des Projekts betreffen. Der Experimentanteil enthält alle Ziele, die der Erfassung von SE-Erfahrungen für die kontinuierliche Verbesserung über Projekte hinweg dienen.

Ziele (QIP2)				
Struktur		Einträge	Ablage	
Projekt	Projektziele	Produktentwicklung	Motivation	C
		Qualitätsanforderungen	GQM-Plan, Qualitätssicherung, Zeitplan, Aufwandsabschätzungen, ...	PM
		Planungsvorgaben	Lebenszyklusmodell, (abstrakter) Projektplan	PM
	Aktivitäten		Grobkonzeption der Tasks	PTEU
	Besprechungsvorbereitungen		ToDo-Listen	V
	Problembeschreibung/ Anforderungsbeschreibung		Ausschnitt aus der Systemdok./Problemlösungsdok.	PTEPS
Lernen	Lernziele		umgangssprachliche Liste	ZE
	Abstraction Sheets		FrameMaker-Dok.	ZE
	GQM-Plan		GQM-Diva oder ViVaGQM	ZE

Projekt

Hierunter werden alle Dokumente gefaßt, welche die Zielsetzung des Projektanteils der Fallstudie beschreiben. Der Projektanteil verfolgt die Entwicklung eines SW-Systems und ist nur an dem zu entwickelnden, eigentlichen Software-Produkt interessiert.

Projekt → Projektziele → Planungsvorgaben

Hier werden alle (High-Level) Planungsmodelle, die Grundlage für den Projektplan sein sollen, angegeben.

Projekt → Problembeschreibung/Anforderungsbeschreibung

Die gesamte Systemdokumentation eines solchen Projekts teilt sich insgesamt in drei Hauptdokumente auf. Der erste dieser Teile ist die Problemlösungsdokumentation. In diesen Teil geht schließlich auch die Problembeschreibung/Anforderungsbeschreibung ein.

Lernen

Alle Dokumente, die experimentelle Zielfindung unterstützen bzw. beschreiben.

Lernen → Lernziele

Dieser Punkt enthält eine umgangssprachlich formulierte Liste der Ziele, die im Projekt verfolgt werden sollten.

2.6 Prozeßwahl/Projektplan (QIP 3)

Der dritte Schritt des QIP enthält alle SE-Erfahrungen, die die Planung der Fallstudie betreffen. Hierbei wird im wesentlichen zwischen der Planung auf Seiten des Managements (s. *Projektplan*) und der Vorbereitung des Entwicklungsprozesses (*Projektunterstützung*) unterschieden. Unter letzterem versteht man z.B. Richtlinien, Wiederverwendungskandidaten oder Templates.

Prozeßwahl/Projektplan (QIP3)					
Struktur		Einträge	Ablage		
Projektplan (Projekt + Lernen)		Gesamtprojektplan (informell, MVP-L, GEM, MoST)	PM		
		Projektmanagement, Produktmanagement, Konfigurationsplan	PM		
		Produktmodell	MPD		
		Prozeßmodell	MPZ		
		Ressourcenmodell	MR		
		Qualitätsmodell	MQ		
		Zeitplan, Gantt-Chart	PM		
		Meßplan	PM		
		Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)	PM		
		TAFs	PTEU		
Projektunterstützung	Richtlinien		Kodeerstellung, Kodeerstellung aus OMT-Entwurf, Komponententesterstellung, Namenskonventionen für NRL-Dokumente, Programmierung	PTEURI	
	Review-Checklisten		Allgemein, Kode, Kunden-Anforderungen, NRL-Dokumente, OMT-Analyse, OMT-Entwurf	PTEURC	
	Beispiele (bei Neuerstellung)		Systemdokumentation, makefiles	PTEPS, PTEUM	
	Templates		Protokolle, (Teile der) Systemdokumentation	PTEUT	
	Wieder-/Weiterverwendung (u. a. bei Änderungen)	Systemdokumentation		(nur Quellen)	PTEPS
		System-Kode		Interface (header), Implementation	PTEPQ
		Makefiles		imake, make, ...	PTEUM
		Tests		Stubs & Treiber	PTEUV
		Werkzeug-Einstellungen		Emacs, FrameMaker, StP/OMT (Tool-Info), ILOG-	PTEUW
	Beschreibung der einzusetzenden Technologien		Technologiepakete zu StP/OMT, RCS, ...	EU	
	Trainingsunterlagen		Tutorials, Einarbeitungsmaterial	V	
	(Externes) Domänenwissen		Gebäudearchitektur-Modell, Zeitreferenzmodell, Datadictionaries für Regelungsalgorithmen	EU	
externe (Literatur-) Referenzen		SCR, [...], ...	EU		

Projektplan

Kann als einzelne Datei (informell und in verschiedenen Spezifikationssprachen) oder zusammengesetzt aus den vorgeschlagenen Teilen eingebracht werden.

Projektunterstützung

Hierunter fallen alle Dokumente, die die Entwicklung (das Team) zur Durchführung des Projekts benötigen.

Projektunterstützung → Richtlinien

Man beachte hierbei insbesondere den Unterschied zwischen Richtlinien zur Kodeerstellung und Richtlinien zur Programmierung. Bei den Richtlinien zur Kodeerstellung handelt es sich um technische Grundlagen, die vor und während der Kodeerstellung beachtet werden sollen; beispielsweise *Filehandling*, Verzeichnisstrukturen oder Benutzung von *Makefiles*. Bei den Programmierrichtlinien handelt es sich hingegen um einen Leitfaden, der während der eigentlichen Kodierung eingehalten werden soll.

Projektunterstützung → Beispiele

Dieser Strukturpunkt enthält Dokumente, die beispielhaft als Vorlage zur Erstellung solcher verwendet werden. Dies scheint insbesondere für Projekte sinnvoll zu sein, in denen die Neuerstellung eines System zentrale Aufgabe ist. (Bei Systemerweiterungen/-änderungen kann die zu erweiternde Systemdokumentation als Beispiel dienen.) Der Grad der notwendigen Anpassung muß/kann durch die Teammitglieder bestimmt werden.

Beispiele hierfür sind: eine Systemdokumentation und makefiles anderer Projekte.

Projektunterstützung → Templates

Im Falle einer völligen Neuerstellung von Dokumenten sind Templates hierzu sinnvoll. (Protokolle werden immer neu erstellt ;-)

Projektunterstützung → Wieder-/Weiterverwendung

Im Falle, daß eine Erweiterung/Änderung eines bestehenden System Ziel des Projekts ist, finden sich hier alle Dokumente, auf denen aufgesetzt werden soll.

Projektunterstützung → Beschreibung der einzusetzenden Technologien &

→ *Domänenwissen &*

→ *Literaturreferenzen*

An dieser Stelle werden Elemente aus dem übergreifenden Bereich referenziert, falls vorhanden. Für zusätzliche SE-Artefakte, welche sich nicht in dem übergreifenden Datenbereich befinden, wird ein temporäres Arbeitsverzeichnis (*Übergreifend*) im Ablagebereich der Fallstudie eingerichtet. SE-Artefakte in diesem Bereich sollten spätestens gegen Ende der Fallstudie in der übergreifenden Bereich übernommen werden.

2.7 Durchführung (QIP4)

Schritt vier des QIP enthält alle Ergebnisse, die während der Durchführung der Fallstudie anfallen. Wir unterscheiden dabei auf oberster Ebene zwischen den Artefakten der Software-Entwicklung (*Entwickeltes SW-System*) und den übrigen Ergebnissen der Projektkontrolle (*Projekttablauf*) inklusive der Meßdatenerfassung.

Durchführung (QIP4)			
Struktur		Einträge	Ablage
Entwickeltes SW-System	Systemdokumentation	gesamt, oder einzeln: Problem-Lösungsdok., SW-Systemdok., SW-Komponentendok., Testfälle,...	PTEPS
	Quelldateien	StP/OMT, C++-Code, Testdateien (Treiber, Stubs, Header-Dateien), makefiles (falls neu erstellt oder überarbeitet)	PTEPQ, PTEUM
	Verifikation	Review-Ergebnisse (der Kunden-Anforderungen, OMT-Analyse, OMT-Entwurf, Kode)	PTEURC
	Validation	Testprotokolle	PTEUV
Projekttablauf	Sitzungsprotokolle	html-Protokolle, FrameMaker, ...	DP
	Meßdaten	Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten	DM
	Prozeßtrace	Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung	DP
	Qualitative Erfahrungen	unstrukturierte "Lessons learned" zur späteren formalen Aufbereitung zu Erfahrungselementen, Informell dokumentierte Entscheidungen, z. B. per email ...	DL

2.8 Analyse (QIP5)

Im fünften Schritt des QIP werden die Ergebnisse der Projektdurchführung analysiert und mit den Planungsdaten verglichen. Dies resultiert meist in überarbeiteten/angepaßten SE-Artefakten der vorangegangenen QIP-Schritte (*Update Modelle*, *Update Projektunterstützung*), in überarbeiteten "Lessons Learned" sowie in einem *Abschlußbericht*.

Analyse (QIP5)			
Struktur		Einträge	Ablage
Update Modelle	Qualitätsmodell	überarbeitete Aufwandsverteilung	PTAE
	Produktmodell	Vorschläge für angepaßte Produktmodelle	PTAE
	Prozeßmodell	Vorschläge für angepaßte Prozeßmodelle	PTAE
	Ressourcenmodell	Vorschläge für angepaßte Ressourcenmodelle	PTAE
Update Projektunterstützung	Richtlinien	überarbeitete Richtlinien	PTAE
	Review-Checklisten	überarbeitete Review-Checklisten	PTAE
	Templates	überarbeitete Templates (FrameMaker, ...)	PTAE
Lessons learned		Projektmanagement, Durchführung, ...	DL
Abschlußbericht		Analyseergebnisse, (kommentierte) Liste der tatsächlich durchgeführten Arbeiten, Vergleich Hypothesen->Meßdaten, Erklärung für Abweichungen, Verbesserungsvorschläge für Folgeprojekte ähnlichen Typs.	PTAE, V

3 Zugriffstruktur für "technische" Praktika (SE-I-Typ)

3.1 "Technische" Praktika

Praktika sind Fallstudien, deren Entwickler sich größtenteils aus Studenten des Studiengangs Informatik zusammensetzen. Sie besitzen meist notwendige Grundkenntnisse jedoch kaum praktische Erfahrung bez. der Entwicklung von Software-Systemen. Desweiteren stehen Studenten nur zu verhältnismäßig unregelmäßigen Zeitpunkten (im Gegensatz zu "professionellen"Entwicklern) zur Verfügung, was die Kommunikation zwischen verschiedenen Entwicklungsteams erschwert.

Im Gegensatz zu Typ "techn. & organ." übernehmen in "technischen" Praktika die Studenten nur die technischen, d.h. Entwicklungs-Rollen, von Anforderungsingenieuren bis hin zu Kodierern und Testern. Die organisatorischen Rollen (Projektplanung und -kontrolle/Management) sind von wissenschaftlichen Mitarbeitern besetzt.

Typische Vertreter dieses Typs von Fallstudie sind SE-I-Praktika der Arbeitsgruppe *Software Engineering*, weswegen diese Praktika auch mit SE-I-Typ benannt werden.

3.2 Strukturbeschreibung

(siehe Einführung in Kapitel 2.2)

3.3 Technische Vorbereitung

Hierunter werden alle Dokumente zusammengefaßt, die vor der eigentlichen Planung des Praktikums erstellt werden müssen bzw. wiederverwendet werden können. Sie dienen der Vorbereitung der Praktikumsplanung.

Technische Vorbereitung			
	Struktur	Einträge	Ablage
Aushänge/Postings	HiWi-Suche	WWW, Aushänge, Newsposting	V
	Externe Aufrufe (zur Nutzung)	email	V
	Praktikumsankündigungen	Aushang, Newsposting	C, V
	Teilnehmerliste (leer)	Tabelle	V
	Erste Besprechung	Aushang	V
Planungstemplates		Abstraction Sheets (leer), Stylefiles für Aushänge, ...	PTEUT,V

Aushänge/Postings

Alle Dokumente zur Suche von HiWis bzw. Suche/Informierung von Studenten.

Planungstemplates

Wiederverwendbare Strukturdokumente zur Vorbereitung der Planung.

3.4 Charakterisierung (QIP1)

Dieser Abschnitt faßt alle SE-Erfahrungen zusammen, die im QIP-Schritt 1 erstellt werden sollten bzw. könnten.

Charakterisierung (QIP1)		
Struktur	Einträge	Ablage
Kontextbeschreibung	Kontextvektor	CV
Teammitglieder	Text-Liste, html-Seite	C
Praktikumsankündigung	Aushang (gekürzt)	C
Beschreibung der einzusetzenden Technologien	→ QIP3	

Dieser Abschnitt faßt alle Dokumente zusammen, die im QIP-Schritt 1 erstellt werden sollten.

Praktikumsankündigung

Der Ausschnitt der Praktikumsankündigung, der das Praktikum inhaltlich charakterisiert.

Beschreibung der einzusetzenden Technologien

Diese werden in QIP3 noch einmal aufgeführt und werden daher an dieser Stelle als Link eingetragen.

3.5 Ziele(QIP2)

Unter diesem Strukturpunkt werden alle Ziele des Praktikums dokumentiert. Hierbei wird zwischen dem Projektanteil (*Projekt*) und dem Experimentanteil (*Lernen*) unterschieden. Der Projektanteil enthält alle Zielsetzungen, die die Entwicklung der geplanten Software sowie die Kontrolle des Projekts betreffen. Der Experimentanteil enthält alle Ziele, die der Erfassung von SE-Erfahrungen für die kontinuierliche Verbesserung über Projekte hinweg dienen.

Ziele (QIP2)				
Struktur		Einträge	Ablage	
Projekt	Projektziele	Produktentwicklung	Motivation, Aushänge (Ausschnitt)	C
		Qualitätsanforderungen	Qualitätsdefinition, GQM-Plan, Zeitrahmen, Aufwandsabschätzungen, ...	PM
		Planungsvorgaben	Lebenszyklusmodell, (abstrakter) Projektplan	PM
	Aufgaben		Grobkonzeption der Aufgabenblätter	PTEU
	Besprechungsvorbereitungen		ToDo-Listen	V
	Problembeschreibung/Anforderungsbeschreibung		Ausschnitt aus der Systemdok./Problemlösungsdok.	PTEPS
Lernen	Lernziele		umgangssprachliche Liste	ZE
	Abstraction Sheets		FrameMaker-Dok.	ZE
	GQM-Plan		GQM-Diva oder ViVaGQM	ZE

Projekt

Hierunter werden alle Dokumente gefaßt, welche die Zielsetzung des Projektanteils des Praktikums beschreiben. Der Projektanteil verfolgt die Entwicklung eines SW-Systems und ist nur an dem eigentlichen Produkt, d.h. dem Software-System, interessiert.

Projekt → Projektziele → Planungsvorgaben

Hier werden alle (High-Level) Planungsmodelle, die Grundlage für den Projektplan sein sollen, angegeben. In Praktikum "SE-1-99" ist dies das SFB-Lebenszyklus-Referenzmodell.

Lernen

Alle Dokumente, die experimentelle Zielfindung unterstützen bzw. beschreiben.

Lernen → Lernziele

Dieser Punkt enthält eine umgangssprachlich formulierte Liste der Ziele, die die Studenten und Betreuer in dem Praktikum verfolgen sollten. Dient als eine Art Referenzliste auch für die am Ende des Praktikums durchgeführte Prüfung.

3.6 Prozeßwahl/Projektplan (QIP3)

Der dritte Schritt des QIP enthält alle SE-Erfahrungen, die die Planung der Fallstudie betreffen. Hierbei wird im wesentlichen zwischen der Planung auf Seiten des Managements (s. *Projektplan*) und der Vorbereitung des Entwicklungsprozesses (*Projektunterstützung*) unterschieden. Unter letzterem versteht man z.B. Richtlinien, Wiederverwendungskandidaten oder Templates.

Prozeßwahl/Projektplan (QIP3)				
Struktur		Einträge	Ablage	
Projektplan (Projekt + Lernen)		Gesamtprojektplan (informell, MVP-L, GEM, MoST)	PM	
		Projektmanagement, Produktmanagement, Konfigurationsplan	PM	
		Produktmodell	MPD	
		Prozeßmodell	MPZ	
		Ressourcenmodell	MR	
		Qualitätsmodell	MQ	
		Zeitplan, Gantt-Chart	PM	
		Meßplan	PM	
Projektunterstützung		Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)	PM	
		Aufgabenblätter	PTEU	
	Richtlinien	Kodeerstellung, Codeerstellung aus OMT-Entwurf, Komponententesterstellung, Namenskonventionen für NRL-Dokumente, Programmierung	PTEURI	
	Review-Checklisten	Allgemein, Kode, Kunden-Anforderungen, NRL-Dokumente, OMT-Analyse, OMT-Entwurf	PTEURC	
	Beispiele (bei Neuerstellung)	Systemdokumentation, makefiles	PTEPS, PTEUM	
	Templates	Protokolle, (Teile der) Systemdokumentation	PTEUT	
	Wieder-/Weiterverwendung (u. a. bei Änderungen)	Systemdokumentation	(nur Quellen)	PTEPS
		System-Kode	Interface (header), Implementation	PTEPQ
		Makefiles	imake, make, ...	PTEUM
		Tests	Stubs & Treiber	PTEUV
		Werkzeug-Einstellungen	Emacs, FrameMaker, StP/OMT (Tool-Info), TeX-Stylefiles	PTEUW
	Beschreibung der einzusetzenden Technologien	Technologiepakete zu StP/OMT, RCS, ...	EU	
	Trainingsunterlagen	Tutorials, Einarbeitungsmaterial	V	
(Externes) Domänenwissen	Gebäudearchitektur-Modell, Zeitreferenzmodell, Datadictionaries für Regelungsalgorithmen	EU		
externe Literaturreferenzen	SCR, [...], ...	EU		

Projektplan

Kann als einzelne Datei oder zusammengesetzt aus den vorgeschlagenen Teilen eingebracht werden.

Projektunterstützung

Hierunter fallen alle Dokumente, die die Entwicklung (Praktikanten) zur Durchführung des Projekts benötigen.

Projektunterstützung → Beispiele

Enthält Dokumente, die beispielhaft als Vorlage zur Erstellung solcher verwendet werden. Dies scheint insbesondere für Projekte sinnvoll zu sein, in denen die Neuerstellung eines System zentrale Aufgabe ist. (Bei Systemerweiterungen/-änderungen dient die zu erweiternde Systemdokumentation als Beispiel.) Der Grad der notwendigen Anpassung muß/kann durch die Studenten bestimmt werden.

Beispiele hierfür sind: ein Systemdokumentation und makefiles anderer Projekte.

Projektunterstützung → Template

Im Falle einer völligen Neuerstellung von Dokumenten sind Templates hierzu sinnvoll. (Protokolle werden immer neu erstellt ;-)

Projektunterstützung → Wieder-/Weiterverwendung

Im Falle, daß eine Erweiterung/Änderung eines bestehenden Systems Ziel des Projekts ist, finden sich hier alle Dokumente, auf denen aufgesetzt werden soll.

Projektunterstützung → Beschreibung der einzusetzenden Technologien &

→ *Domänenwissen &*

→ *Literaturreferenzen*

An dieser Stelle werden Elemente aus dem übergreifenden Bereich referenziert, falls vorhanden. Für zusätzliche SE-Artefakte, welche sich nicht in dem übergreifenden Datenbereich befinden, wird ein temporäres Arbeitsverzeichnis (*Übergreifend*) im Ablagebereich des Praktikums eingerichtet. SE-Artefakte in diesem Bereich sollten spätestens gegen Ende der Fallstudie in der übergreifenden Bereich übernommen werden.

3.7 Durchführung (QIP4)

Schritt vier des QIP enthält alle Ergebnisse, die während der Durchführung des Praktikums anfallen. Wir unterscheiden dabei auf oberster Ebene zwischen den Artefakten der Software-Entwicklung (*Entwickeltes SW-System*) und den übrigen Ergebnissen der Projektkontrolle (*Projekttablauf*) inklusive der Meßdatenerfassung.

Durchführung (QIP4)			
Struktur		Einträge	Ablage
Entwickeltes SW-System	Systemdokumentation	gesamt, oder einzeln: Problem-Lösungsdok., SW-Systemdok., SW-Komponentendok., Testfälle,...	PTEPS
	Quelldateien	StP/OMT, C++-Kode, Testdateien (Treiber, Stubs, Header-Dateien), makefiles (falls neu erstellt oder überarbeitet)	PTEPQ, PTEUM
	Verifikation	Review-Ergebnisse (der Kunden-Anforderungen, OMT-Analyse, OMT-Entwurf, Kode)	PTEURC
	Validation	Testprotokolle	PTEUV
Projekttablauf	Sitzungsprotokolle	html-Protokolle, FrameMaker, ...	DP
	Meßdaten	Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten	DM
	Prozeßtrace	Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung	DP
	Qualitative Erfahrungen	unstrukturierte Lessons learned zur späteren formalen Aufbereitung zu Erfahrungselementen, Informell dokumentierte Entscheidungen, z.B. per email ...	DL

3.8 Analyse (QIP5)

Im fünften Schritt des QIP werden die Ergebnisse der Projektdurchführung analysiert und mit den Planungsdaten verglichen. Dies resultiert meist in überarbeiteten/angepaßten SE-Artefakten der vorangegangenen QIP-Schritte (*Update Modelle*, *Update Projektunterstützung*), in überarbeiteten "Lessons Learned" sowie in einem *Abschlußbericht*.

Analyse (QIP5)			
Struktur		Einträge	Ablage
Update Modelle	Qualitätsmodell	überarbeitete Aufwandsverteilung	PTAE
	Produktmodell	Vorschläge für angepaßte Produktmodelle	PTAE
	Prozeßmodell	Vorschläge für angepaßte Prozeßmodelle	PTAE
	Ressourcenmodell	Vorschläge für angepaßte Ressourcenmodelle	PTAE
Update Projektunterstützung	Richtlinien	überarbeitete Richtlinien	PTAE
	Review-Checklisten	überarbeitete Review-Checklisten	PTAE
	Templates	überarbeitete Templates (FrameMaker, ...)	PTAE
Lessons learned		Projektmanagement, Durchführung, ...	DL
Abschlußbericht		Analyseergebnisse, (kommentierte) Liste der tatsächlich durchgeführten Arbeiten, Vergleich Hypothesen <-> Meßdaten, Erklärung für Abweichungen, Verbesserungsvorschläge für Folgeprojekte ähnlichen Typs	PTAE, V

4 Zugriffstruktur für "technische & organisatorische" Praktika (SE-II-Typ)

4.1 "Technische & organisatorische" Praktika

Praktika sind Fallstudien, deren Entwickler sich größtenteils aus Studenten des Studiengangs Informatik zusammensetzen. Sie besitzen meist notwendige Grundkenntnisse jedoch kaum praktische Erfahrung bez. der Entwicklung von Software-Systemen. Desweiteren stehen Studenten nur zu verhältnismäßig unregelmäßigen Zeitpunkten (im Gegensatz zu "professionellen"Entwicklern) zur Verfügung, was die Kommunikation zwischen verschiedenen Entwicklungsteams erschwert.

Der *SE-II*-Typ des Fallstudientyps "Praktika" zeichnet sich zusätzlich dadurch aus, daß die Praktikanten ebenfalls die Planung ihres Projekts mitgestalten und die Projektkontrolle/-management ausüben, d.h. auch organisatorische Rollen übernehmen. Sie durchlaufen den gesamten QIP-Zyklus, indem sie verschiedene Rollen annehmen. Um dem Rechnung zu tragen, sind in QIP4 (Durchführung des Praktikums) Links zu einem Subprojekt eingetragen, welches die gesamte Arbeit der Praktikanten aufnehmen soll.

Typische Vertreter dieses Typs von Fallstudie sind SE-II-Praktika der Arbeitsgruppe *Software Engineering*, weswegen diese Praktika auch mit SE-II-Typ benannt werden.

4.2 Strukturbeschreibung

(siehe Einführung in Kapitel 2.2)

4.3 Technische Vorbereitung

Hierunter werden alle Dokumente zusammengefaßt, die vor der eigentlichen Planung des Praktikums erstellt werden müssen bzw. wiederverwendet werden können. Sie dienen der Vorbereitung der Praktikumsplanung.

Technische Vorbereitung			
	Struktur	Einträge	Ablage
Aushänge/Postings	HiWi-Suche	WWW, Aushänge, Newsposting	V
	Externe Aufrufe (zur Nutzung)	email	V
	Praktikumsankündigungen	Aushang, Newsposting	C,V
	Teilnehmerliste (leer)	Tabelle	V
	Erste Besprechung	Aushang	V
Planungstemplates		Abstraction Sheets (leer), Stylefiles für Aushänge, ...	PTEUT, V

Dieser Punkt ist identisch mit 'Technische Vorbereitung' des SE-I-Typs (s. Seite 13)

4.4 Charakterisierung (QIP1)

Dieser Abschnitt faßt alle SE-Erfahrungen zusammen, die im QIP-Schritt 1 erstellt werden sollten bzw. könnten.

Charakterisierung (QIP1)		
Struktur	Einträge	Ablage
Kontextbeschreibung	Kontextvektor	CV
Teammitglieder	Text-Liste, html-Seite	C
Praktikumsankündigung	Aushang (gekürzt)	C
Beschreibung der einzusetzenden Technologien	→ QIP3	

Dieser Punkt ist identisch mit QIP1 des SE-I-Typs (s. Seite 14)

4.5 Ziele (QIP2)

Unter diesem Strukturpunkt werden alle Ziele des Praktikums dokumentiert. Hierbei wird zwischen dem Projektanteil (*Projekt*) und dem Experimentanteil (*Lernen*) unterschieden. Der Projektanteil enthält alle Zielsetzungen, die die Entwicklung der geplanten Software sowie die Kontrolle des Projekts betreffen. Der Experimentanteil enthält alle Ziele, die der Erfassung von SE-Erfahrungen für die kontinuierliche Verbesserung über Projekte hinweg dienen.

Ziele (QIP2)				
Struktur		Einträge	Ablage	
Projekt	Projektziele	Produktentwicklung	Motivation, Aushänge (Ausschnitt)	C
		Qualitätsanforderungen	GQM-Plan, Zeitplan, ...	PM
	Aufgaben	Grobkonzeption der Aufgabenblätter	PTEU	
	Besprechungsvorbereitungen	ToDo-Listen	V	
	Problembeschreibung/Anforderungsbeschreibung	Planung (Musterlösung) oder Vorgabe (s. u.)	PTEPS	
Lernen	Lernziele	umgangssprachliche Liste	ZE	
	Abstraction Sheets	FrameMaker-Dok.	ZE	
	GQM-Plan	GQM-Diva, ViVaGQM	ZE	

Projekt → *Problembeschreibung/Anforderungsbeschreibung*

Dieser Abschnitt beschreibt die Ziele der technischen Rollen für das Projekt. Es enthält eine Erweiterung des zu grundlegenden Systems (s. 4.6 (QIP3), Projektunterstützung → Wieder-/Weiterverwendung → Systemdokumentation). Diese Zielbeschreibung ist mit der Problembeschreibung/Anforderungsbeschreibung des zu grundlegenden Systems (s. 4.1.2.4 (QIP3) und des Sub-QIP2 (s. 4.7.1.1, s. Seite 24) identisch, wenn von den Studenten keine eigenständige Erweiterung/Änderung der Anforderungsbeschreibung verlangt wird.

Beispielsweise könnte die Änderung der Anforderung als Aufgabe formuliert sein. In diesem Fall stelle diese Anforderungsbeschreibung eine Erweiterung der in QIP3 vorgegebenen Anforderungsbeschreibung dar (eine Art Musterlösung), welche die Studenten als ihre eigene Lösung in Sub-QIP2 und Sub-QIP4 eintragen.

Lernen

Alle unter diesem Punkt zusammengefaßten Punkte enthalten nur Lernziele über die Planung und Durchführung des Praktikums (d.h. aus Sicht des Betreuers). Ziele der Planung und Durchführung erstellen die Studenten selbst, daher sind diese Ziele unter QIP4 zu finden.

Lernen→Abstraction Sheets

Für das Ausfüllen der Abstraction Sheets auf dieser Ebene können die Studenten nach Hypothesen (die Art und Weise der Durchführung) befragt werden. Sie werden als Vorbereitung für das Praktikum erstellt. (vgl. auch Sub-QIP2)

4.6 Prozeßwahl/Projektplan (QIP3)

Der dritte Schritt des QIP enthält alle SE-Erfahrungen, die die Planung der Fallstudie betreffen. Hierbei wird im wesentlichen zwischen der Planung auf Seiten des Managements (s. *Projektplan*) und der Vorbereitung des Entwicklungsprozesses (*Projektunterstützung*) unterschieden. Unter letzterem versteht man z.B. Richtlinien, Wiederverwendungskandidaten oder Templates.

Man beachte, daß die Planung auch eine Art Meta-Planung für die Studenten enthält, d.h. sie enthält Artefakte, die beschreiben, wie die Planung der Studenten (Sub-Projekt) ablaufen soll.

Prozeßwahl/Projektplan (QIP3)				
Struktur		Einträge	Ablage	
Projektplan (Projekt + Lernen)		Gesamtprojektplan (informell, MVP-L, GEM, MoST)	PM	
		Projektmanagement, Produktmanagement, Konfigurationsplan	PM	
		Produktmodell	MPD	
		Prozeßmodell	MPZ	
		Ressourcenmodell	MR	
		Qualitätsmodell	MQ	
		Zeitplan, Gantt-Chart	PM	
		Meßplan	PM	
		Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)	PM	
		Aufgabenblätter	PTEU	
Projektunterstützung	Richtlinien		Kodeerstellung, Komponententeststellung, Namenskonventionen für NRL-Dokumente, Programmierung	PTEURI
	Review-Checklisten		Allgemein, Code, Kunden-Anforderungen, NRL-Dokumente, OMT-Analyse, OMT-Entwurf	PTEURC
	Beispiele (u. a. bei Neuerstellung)		Systemdokumentation, makefiles, Projektplan, GQM-Plan	PTEPS, PTEUM, PM, ZE
	Templates		Protokolle, Projektplan, Abstraction Sheets (→ technische Vorbereitung)	PTEUT, PM
	Wieder-/Wiederverwendung (u. a. bei Änderungen)	Planung	Projektplan, Lebenszyklusmodell	PM
		Systemdokumentation	(nur Quellen)	PTEPS
		System-Kode	Interface (header), Implementation	PTEPQ
		Makefiles	imake, make, ...	PTEUM
		Tests	Stubs & Treiber	PTEUV
		Werkzeug-Einstellungen	Emacs, FrameMaker, StP/OMT (Tool-Info), TeX-Stylefiles	PTEUW
	Projektplan		zu erweiternder Projektplan	PM
	Beschreibung der einzusetzenden Technologien		Technologiepakete zu StP/OMT, RCS, ...	EU
	Trainingsunterlagen		Tutorials, Einarbeitungsmaterial	V
externe Literaturreferenzen		SCR, [...], ...		

Projektplan

Der Projektplan enthält:

- Planung der Planungsphasen des Praktikums (d.h. des Subprojektes)
- Planung der Durchführungsphase des Subprojektes als eine Art Musterlösung.

Projektunterstützung

Hierunter fallen alle Dokumente, die die Entwicklung (Praktikanten) zur Durchführung des Projekts benötigen.

Projektunterstützung → Beispiele

Enthält unter anderem einen (vollständigen) beispielhaften Projektplan und Abstraction Sheets, falls diese in den folgenden Abschnitten nicht vorhanden sind:

Projektunterstützung → Template

Projektunterstützung → Wieder-/Weiterverwendung → Projektplan

Projektunterstützung → Template

Enthält unter anderem einen leeren Projektplan, der als Start-Dokument verwendet werden kann. Ebenso kann ein Projektplan unter den folgenden Abschnitten als Template verwendet werden:

Projektunterstützung → Beispiele

Projektunterstützung → Wieder-/Weiterverwendung → Projektplan

Projektunterstützung → Wieder-/Weiterverwendung → Projektplan

Enthält einen zum Teil vollständigen Projektplan, der als Teil der Praktikumsaufgaben erweitert/geändert/vervollständigt werden soll.

4.7 Durchführung (QIP4)

Schritt vier des QIP enthält alle Ergebnisse, die während der Durchführung der Fallstudie anfallen. Wir unterscheiden dabei auf oberster Ebene zwischen den Ergebnissen des *Subprojekts*, welches vollständig von den Studenten übernommen wird, und den Ergebnissen der Projektüberwachung (*Projekttablauf*) inklusive der Meßdatenerfassung. Diese enthält auch Meßdaten über die Qualität der Sub-Projekt-Planung und -kontrolle.

Durchführung (QIP4)			
	Struktur	Einträge	Ablage
→ Subprojekt	QIP2	→ Sub-QIP2	(s. u.)
	QIP3	→ Sub-QIP3	(s. u.)
	QIP4	→ Sub-QIP4	(s. u.)
	QIP5	→ Sub-QIP5	(s. u.)
Projekttablauf	Sitzungsprotokolle	html-Protokolle, FrameMaker, ...	DP
	Meßdaten	Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten	DM
	Prozeßtrace	Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung	DP
	Qualitative Erfahrungen	unstrukturierte Lessons learned zur späteren formalen Aufbereitung zu Erfahrungselementen	DL

Subprojekt

Die Durchführungsphase des Praktikums besteht aus der Planung (Sub-QIP1-3) und der Durchführung (Sub-QIP4) des Subprojektes. Sub-QIP1 entspricht QIP1, und wird daher hier nicht mehr aufgeführt.

Diese Überschrift wird zusätzlich als Link auf das gesamte Subprojekt (Sub-QIP1-5) realisiert.

Subprojekt → Sub-QIP2 - QIP5

Diese Punkte werden im Subprojekt in Kapitel 4.7.1 (s. Seite 24) erläutert.

Projektlauf → Sitzungsprotokolle

Enthält Protokolle **aller** Sitzungen.

Projektlauf → Meßdaten

Enthält alle Meßdaten, zu denen in QIP2 und QIP3 Ziele und Meßprogramm aufgestellt wurden.

Projektlauf → Qualitative Erfahrungen

Was wurde über die Organisation und Planung des Praktikums (Betreuer) gelernt; Aussagen von Betreuern und Praktikanten. Dienen den Betreuern/ der Analyse zur Verbesserung zukünftiger Praktika.

4.7.1 Subprojekt, SE-II-Typ

4.7.1.1 Ziele (Sub-QIP2)

Ziele (Sub-QIP2)			
Struktur		Einträge	Ablage
Projekt	Problembeschreibung/Anforderungsbeschreibung	Ausschnitt aus der Systemdokumentation (Sub-QIP4)	PTEPS
Lernen	Lernziele	umgangssprachliche Liste	ZE
	Abstraction Sheets	(ausgefüllt)	ZE
	GQM-Plan	GQM-Diva, ViVaGQM	ZE

Projekt → Problembeschreibung/...

Verweist auf die Problembeschreibung, die Grundlage für die Änderung des zu erstellenden Systems ist. Ist u.U. identisch mit dem gleichnamigen Eintrag unter 4.5, QIP2 Hauptprojekt (s. Seite 20).

Lernen → Abstraction Sheets

Diese *Abstraction Sheets* werden im Rahmen der Praktikumsaufgaben ausgefüllt (von den Praktikanten mit Unterstützung der Betreuer). (vgl. auch QIP2, Hauptprojekt (s. Seite 20))

4.7.1.2 Prozeßwahl/Projektplan (Sub-QIP3)

Prozeßwahl/Projektplan (Sub-QIP3)		
Struktur	Einträge	Ablage
Projektplan (Projekt + Lernen)	Gesamtprojektplan (informell, MVP-L, GEM, MoST)	PM
	Projektmanagement, Produktmanagement, Konfigurationsplan	PM
	Produktmodell	MPD
	Prozeßmodell	MPZ
	Ressourcenmodell	MR
	Qualitätsmodell, Zeitplan, Gantt-Chart	MQ, PM
	Meßplan	PM
	Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)	PM

Projektplan

Der von den Praktikanten entwickelte Projektplan.

Projektunterstützung

Dieser Punkt fehlt im Gegensatz zum Hauptprojekt, da im Praktikum selbst die Projektunterstützung aus dem Hauptprojekt zur Anwendung kommt und keine zusätzliche erstellt wird.

4.7.1.3 Durchführung (Sub-QIP4)

Schritt vier des QIP für das Sub-Projekt enthält alle Ergebnisse, die während der Durchführung des Sub-Projekts anfallen. Wir unterscheiden dabei wieder auf oberster Ebene zwischen den Artefakten der Software-Entwicklung (*Entwickeltes SW-System*) und den übrigen Ergebnissen der Projektkontrolle (*Projekt-Ablauf*) inklusive der Meßdatenerfassung.

Durchführung (Sub-QIP4)			
Struktur	Einträge	Ablage	
Entwickeltes SW-System	Systemdokumentation	gesamt, oder einzeln: Problem-Lösungsdok., SW-Systemdok., SW-Komponentendok., Testfälle,...	PTEPS
	Quelldateien	StP/OMT, C++-Kode, Testdateien (Treiber, Stubs, Header-Dateien), makefiles (falls neu erstellt oder überarbeitet)	PTEPQ, PTEUM
	Verifikation	Review-Ergebnisse (der Kunden-Anforderungen, OMT-Analyse, OMT-Entwurf, Kode)	PTEURC
	Validation	Testprotokolle	PTEUV
Projektablauf	Sitzungsprotokolle	→ Hauptprojekt QIP4	
	Meßdaten	Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten	DM
	Prozeßtrace	Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung	DP
	Qualitative Erfahrungen	unstrukturierte Lessons learned zur späteren formalen Aufbereitung zu Erfahrungselementen	DL

Projektablauf → Meßdaten

Enthält alle Meßdaten, zu denen in Sub-QIP2 und Sub-QIP3 Ziele und Meßprogramm aufgestellt wurden.

Projektablauf → Qualitative Erfahrungen

Welche Lehren ziehen die Praktikanten aus dem Praktikum?

4.7.1.4 Analyse (Sub-QIP5)

Im fünften Schritt des QIP für das Sub-Projekt werden die Ergebnisse der Projektdurchführung analysiert und mit den Planungsdaten verglichen. Dies resultiert meist in überarbeiteten/angepaßten SE-Artefakten der vorangegangenen QIP-Schritte (*Update Modelle, Update Projektunterstützung*), in überarbeiteten "Lessons Learned".

Diese Artefakte werden ebenfalls von den Studenten erstellt, falls die Analyse des Sub-Projekts in der Aufgabenstellung enthalten ist. Allerdings dient dieser Schritt nur der Dokumentation des letzten Lernschritts der Studenten. Basis für den experiment-übergreifenden Datenbereich ist dieser Abschnitt jedoch nicht. Zur Einlagerung in den übergreifenden Bereich werden die Daten aus Kapitel 4.8 herangezogen.

Analyse (Sub-QIP5)			
Struktur		Einträge	Ablage
Update Modelle	Qualitätsmodell	überarbeitete Aufwandsverteilung	PTAE
	Produktmodell	Vorschläge für angepaßte Produktmodelle	PTAE
	Prozeßmodell	Vorschläge für angepaßte Prozeßmodelle	PTAE
	Ressourcenmodell	Vorschläge für angepaßte Ressourcenmodelle	PTAE
Lessons learned		Projektmanagement, Durchführung, ...	DL

4.8 Analyse (QIP5)

Im fünften Schritt des QIP werden die Ergebnisse der Projektdurchführung analysiert und mit den Planungsdaten verglichen. Hier gehen auch Ergebnisse bez. der Sub-Projekt-Planung und -durchführung ein soweit sie relevant für eine zukünftige Verbesserung des Praktikums sein können. Das Ergebnis der Analyse besteht aus überarbeiteten/angepaßten SE-Artefakten der vorangegangenen QIP-Schritte (*Update Modelle, Update Projektunterstützung*), in überarbeiteten "Lessons Learned" sowie in einem *Abschlußbericht*.

Analyse (QIP5)			
Struktur		Einträge	Ablage
Update Modelle	Qualitätsmodell	überarbeitete Aufwandsverteilung	PTAE
	Produktmodell	Vorschläge für angepaßte Produktmodelle	PTAE
	Prozeßmodell	Vorschläge für angepaßte Prozeßmodelle	PTAE
	Ressourcenmodell	Vorschläge für angepaßte Ressourcenmodelle	PTAE
Update Projektunterstützung	Richtlinien	überarbeitete Richtlinien	PTAE
	Review-Checklisten	überarbeitete Review-Checklisten	PTAE
	Templates	überarbeitete Templates (FrameMaker, ...)	PTAE
Lessons learned		Projektmanagement, Durchführung, ...	DL
Abschlußbericht		Analyseergebnisse, (kommentierte) Liste der tatsächlich durchgeführten Arbeiten, Vergleich Hypothesen <->Meßdaten, Erklärung für Abweichungen, Verbesserungsvorschläge für Folgeprojekte ähnlichen Typs	PTAE, V

5 Gegenüberstellung der Zugriffstrukturen für Fallstudien

5.1 Allgemein

Im folgenden Kapitel werden die drei Fallstudientypen (*Team-Projekte*, "*technische*" *Praktika* und "*techn. & organ.*" *Praktika*) gegenübergestellt. Es wird hervorgehoben, welche Einträge zwischen einigen Fallstudientypen gemeinsam sind und in welchen Strukturelementen und Inhalten sie sich unterscheiden. Die Gegenüberstellung wird ebenfalls mit Tabellen beschrieben.

In den ersten Spalten der Tabelle (von links) sind dabei gemeinsame Strukturmerkmale der drei Typen festgehalten (Titel: *Gemeinsame Struktur*). In den Spalten rechts wird dann schließlich nach dem jeweiligen Fallstudientyp differenziert.

Die hell schattierten (■) Felder beinhalten Strukturelemente, die bereits im leeren Zugriffsschema fest vorgegeben sind. Die weißen Felder (□) stellen (Vorschläge für) Einträge dar, die manuell eingetragen werden können/müssen, falls Inhalte dieser Art vorhanden sind. Die Inhalte dieser Felder müssen nicht vollständig sein. Diese sind vollständig aus den ausführlicheren Beschreibungen der drei Fallstudientypen entnommen.

Die dunkel schattierten (■) Felder zeigen an, daß es zu den jeweiligen Einträgen/Strukturelementen eines anderen Fallstudientyps keine Pendanten zu dem Typ, in dessen Spalte dieses Feld ist, gibt.

Zur Vermeidung von doppelten Einträgen sind —ebenso wie bei den einzelnen Beschreibungen der Fallstudientypen— Links (markiert mit → <Ziel>) auf andere Abschnitte eingetragen. Auch findet man hier in der rechten Spalte die Angabe für das Verzeichnis der Ablagestruktur.

In der rechten Spalte sind wieder die Ablageverzeichnisse vermerkt. Sind für alle Fallstudientypen die Ablageverzeichnisse gleich, so ist genau ein Eintrag vorhanden. Unterschiede sind wie folgt notiert:

- durch "/": Die unterschiedlichen Fallstudientypen werden durch ein "/" getrennt. Es können also maximal zwei Trenner auftreten. Die Elemente vor, zwischen und nach diesem Trenner können noch einmal differenziert werden
- durch ",": Dies beschreibt die unterschiedliche Ablage für Elemente innerhalb eines Fallstudientyps. Die Ablageverzeichnisse werden in der Reihenfolge der entsprechenden Einträge angegeben.

Erläuterungen zu den einzelnen QIP-Schritten bzw. zu den unterschiedlichen Einträgen wurden in diesem Kapitel nicht noch einmal vorgenommen. Diese können den vorangegangenen Kapiteln entnommen werden.

5.2 Technische Vorbereitung

Gemeinsame Struktur		Struktur () und Inhalte ()			Ablage	
		Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika		
Technische Vorbereitung	Aushänge/Postings	Externe Aufrufe (zur Nutzung)			V	
		email				
		Erste Besprechung(en)			V / V	
		email, WWW	Aushang			
		Teammitglieder-Suche	HiWi-Suche		V	
		WWW, Aushänge, Newsposting				
		Verpflichtung von HiWis / AG-Mitarbeitern				V
				Praktikumsankündigungen		C, V
				Aushang, Newsposting		
				Teilnehmerliste (leer)		V
		Tabelle				
Technische Vorbereitung	Planungstemplates			PTEUT		
	Abstraction Sheets (leer)					
			Stylefiles für Aushänge, ...		V	

5.3 Charakterisierung (QIP1)

Gemeinsame Struktur		Struktur () und Inhalte ()			Ablage
		Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika	
Charakterisierung (QIP1)	Kontextbeschreibung			CV	
	Kontextvektor				
	Teammitglieder			C	
	Text-Liste, html-Seite				
	Projektankündigung	Praktikumsankündigung		C / C	
	Aushang, WWW	Aushang (gekürzt)			
	Beschreibung der einzusetzenden Technologien				
→ QIP3					

5.4 Ziele (QIP2)

Gemeinsame Struktur		Struktur () und Inhalte ()			Ablage	
		Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika		
Ziele (QIP2)	Projekt	Projektziele	Produktentwicklung			
			Motivation			C
			Aushänge (Ausschnitt)			C
			Qualitätsanforderungen			
			Qualitätsdefinition, GQM-Plan, Zeitrahmen, Aufwandsabschätzungen, ...			PM
			Planungsvorgaben			
	Lebenszyklusmodell, (abstrakter) Projektplan			PM, PM		
	Projekt		Aktivitäten	Aufgaben		
			Grobkonzeption der Tasks	Grobkonzeption der Aufgabenblätter		PTEU
			Besprechungsvorbereitungen			
			ToDo-Listen			V
			Problembeschreibung/Anforderungsbeschreibung			
			Ausschnitt aus der Systemdok./Problemlösungsdok.		Planung (Musterlösung) oder Vorgabe	PTEPS
	Lernen		Lernziele			
			umgangssprachliche Liste			ZE
			Abstraction Sheets			
			FrameMaker-Dok.			ZE
			GQM-Plan			
GQM-Diva oder ViVaGQM			ZE			

5.5 Prozeßwahl/Projektplan (QIP3)

Gemeinsame Struktur	Struktur () und Inhalte ()			Ablage
	Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika	
Prozeßwahl/Projektplan (QIP3)	Projektplan (Projekt + Lernen)			
	Gesamtprojektplan (informell, MVP-L, GEM. MoST)			PM
	Projektmanagement, Produktmanagement, Konfigurationsplan			PM
	Produktmodell			MPD
Prozeßmodell			MPZ	
Ressourcenmodell			MR	
Qualitätsmodell			MQ	
Zeitplan, Gantt-Chart			PM	
Meßplan			PM	
Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)			PM	
TAFs		Aufgabenblätter		PTEU
Prozeßwahl/Projektplan (QIP3)	Projektunterstützung	Richtlinien		
		Kodeerstellung, Kodeerstellung aus OMT-Entwurf, Komponententesterstellung, Namenskonventionen für NRL-Dokumente, Programmierung		PTEURI
		Review-Checklisten		
Allgemein, Kode, Kunden-Anforderungen, NRL-Dokumente, OMT-Analyse, OMT-Entwurf			PTEURC	

Gemeinsame Struktur		Struktur () und Inhalte ()			Ablage	
		Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika		
Prozeßwahl/Projektplan (QIP3)	Projektunterstützung	Beispiele (bei Neuerstellung)		Beispiele (u.a. bei Neuerstellung)	PTEPS, PTEUM	
		Systemdokumentation, makefiles				Projektplan, GQM-Plan
					PM, ZE	
		Templates			PTEUT	
		Protokolle, (Teile der) Systemdokumentation				
				Projektplan, Abstraction Sheets (→ technische Vorbereitung)	PM, PTEUT	
	Projektunterstützung	Wieder-/Weiterverwendung (u.a. bei Änderungen)	Systemdokumentation			PTEPS
			(nur Quellen)			
			System-Kode			PTEPQ
			Interface (header), Implementation			
			Makefiles			PTEUM
			imake, make, ...			
			Tests			PTEUV
			Stubs & Treiber			
			Werkzeug-Einstellungen			PTEUW
			Emacs, FrameMaker, StP/OMT (ToolInfo)			
			ILOG-Views		TeX-Stylefiles	
				Projektplan	PM	
				zu erweiternder Projektplan		
	Projektunterstützung	Beschreibung der einzusetzenden Technologien			EU	
		Technologiepakete zu StP/OMT, RCS, ...				
		Trainingsunterlagen			V	
		Tutorials, Einarbeitungsmaterial				
		(externes) Domänenwissen			EU	
		Gebäudearchitektur-Modell, Zeitreferenzmodell, Datadictionaries für Regelungsalgorithmen				
	externe Literaturreferenzen			EU		
	SCR, [...], ...					

5.6 Durchführung (QIP4)

Gemeinsame Struktur		Struktur (■) und Inhalte (□)			Ablage	
		Team-Projekte	"Technische" Praktika	"Techn. & org." Praktika		
Durchführung (QIP 4)	Entwickeltes SW-System	Systemdokumentation				
		gesamt, oder einzeln: Problem-Lösungsdok., SW-Systemdok., SW-Komponentendok., Testfälle,...				PTEPS
		Quelldateien				PTEPQ, PTEUM
		StP/OMT, C++-Kode, Testdateien (Treiber, Stubs, Header-Dateien), makefiles (falls neu erstellt oder überarbeitet)				
		Verifikation				PTEURC
		Review-Ergebnisse (der Kunden-Anforderungen, OMT-Analyse, OMT-Entwurf, Kode)				
		Validation				PTEUV
Testprotokolle						

Gem. Strukt.	Struktur (■) und Inhalte (□)				Ablage					
	Team-Proj.	Tech. Prakt.	"Techn. & org." Praktika							
Durchführung (QIP 4)				Subprojekt ↑ Subprojekt						
						Ziele (Sub-QIP2)	Projekt	Problembeschreibung/Anforderungsbeschreibung	PTEPS	
								Ausschnitt aus der Systemdokumentation (Sub-QIP4)		
						Ziele (Sub-QIP2)	Lernen	Lernziele	ZE	
								umgangssprachliche Liste		
								Abstraction Sheets	ZE	
								(ausgefüllt)		
						Ziele (Sub-QIP2)	Lernen	GQM-Plan	ZE	
								GQM-Diva oder ViVa-GQM		
								Prozesswahl/Projektplan (Sub-QIP3)		Projektplan (Projekt + Lernen)
Projektmanagement, Produktmanagement, Konfigurationsplan	PM									
Produktmodell	MPD									
Prozeßmodell	MPZ									
Ressourcenmodell	MR									
Qualitätsmodell, Zeitplan, Gantt-Chart	MQ, PM									
Meßplan	PM									
Technik zur Erfassung der Meßdaten: (leere) Fragebögen (Aufwand, Fehler, Fehlerklassen, Kalenderzeit, Produktcharakteristika)		PM								

Gem. Strukt.	Struktur (■) und Inhalte (□)			Ablage	
	Team-Proj.	Tech. Prakt.	"Techn. & org." Praktika		
Durchführung (QIP4)	■	■	Subprojekt → Durchführung (Sub-QIP4)	Systemdokumentation gesamt, oder einzeln: Problem-Lösungsdok., SW-Systemdok., SW-Komponentendok., Testfälle,...	PTEPS
				Quelldateien StP/OMT, C++-Kode, Testdateien (Treiber, Stubs, Header-Dateien), makefiles (falls neu erstellt oder überarbeitet)	PTEPQ, PTEUM
				Verifikation Review-Ergebnisse (der Kunden-Anforderungen, OMT-Analyse, OMT-Entwurf, Kode)	PTEURC
				Validation Testprotokolle	PTEUV
				Sitzungsprotokolle → Hauptprojekt QIP4	
				Meßdaten Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten	DM
				Prozeßtrace Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung	DP
				Qualitative Erfahrungen unstrukturierte Lessons learned zur späteren formalen Aufbereitung zu Erfahrungselementen, ...	DL
				Qualitätsmodell überarbeitete Aufwandsverteilung	PTAE
				Produktmodell Vorschläge für angepaßte Produktmodelle	PTAE
				Prozeßmodell Vorschläge für angepaßte Prozeßmodelle	PTAE
				Ressourcenmodell Vorschläge für angepaßte Ressourcenmodelle	PTAE
				Lessons learned Projektmanagement, Durchführung	DL

Gem. Strukt.	Struktur (■) und Inhalte (□)			Ablage
	Team-Proj.	Tech. Prakt.	"Techn. & org." Praktika	
Durchführung (QIP4))	Projektablauf	Sitzungsprotokolle		DP
		html-Protokolle, FrameMaker, ...		
		Meßdaten		DM
		Aufwandsdaten (absolut, absolut nach Phasen, relativ ohne/mit Projektmanagement, relativ ohne/mit Projektmanagement nach Phasen), Fehlerdaten, Effektivitätsdaten		
		Prozeßtrace		DP
		Gantt-Grafik, Beschreibung in Protokollen, tabellarischer Prozeßtrace, Abweichungen von der Projektplanung, Abweichungen von der Kalenderzeitplanung		
		Qualitative Erfahrungen		DL
unstrukturierte Lessons learned zur späteren formalen Aufbereitung zu Erfahrungselementen, Informell dokumentierte Entscheidungen, z.B. per email ...				

5.7 Analyse (QIP5)

Gem. Struktur	Struktur (■) und Inhalte (□)			Ablage
	Team-Projekte	"Technische"Praktika	"Techn. & org." Praktika	
Analyse (QIP5)	Update Modelle	Qualitätsmodell		PTAE
		überarbeitete Aufwandsverteilung		
		Produktmodell		PTAE
		Vorschläge für angepaßte Produktmodelle		
		Prozeßmodell		PTAE
		Vorschläge für angepaßte Prozeßmodelle		
		Ressourcenmodell		PTAE
	Vorschläge für angepaßte Ressourcenmodelle			
	Update Projektunterstützung	Richtlinien		PTAE
		überarbeitete Richtlinien		
		Review-Checklisten		PTAE
		überarbeitete Review-Checklisten		
		Templates		PTAE
	überarbeitete Templates (FrameMaker, ...)			
Analyse (QIP5)	Lessons learned		DL	
	Projektmanagement, Durchführung, ...			
	Abschlußbericht		PTAE, V	
Analyseergebnisse, (kommentierte) Liste der tatsächlich durchgeführten Arbeiten, Vergleich Hypothesen ↔ Meßdaten, Erklärung für Abweichungen, Verbesserungsvorschläge für Folgeprojekte ähnlichen Typs				

6 Ablagestruktur für Fallstudien

6.1 Allgemein

Bei der Ablagestruktur handelt es sich um eine Struktur zur Datenablage gleichermaßen für Fallstudien und kontrollierte Experimente, weswegen im Verlauf dieses Kapitels auch auf die Inhalte der untersten Strukturebenen für Fallstudien zumindest kurz eingegangen wird. Die Einheitlichkeit der Struktur wird durch eine grobe Granularität der Strukturpunkte erkauft. In den untersten Ebenen der Struktur können daher —wie auf der Zugriffstruktur— weitere Ebenen zur Strukturierung eingefügt werden, falls dies notwendig erscheint. Falls auf den unteren Ebenen regelmäßig Erweiterungen angebracht zu sein scheinen, ist dies ein Hinweis für eine Übernahme in die feste Struktur. Die zusätzlichen, manuell eingefügten Ebenen werden durch deren Schreibweise von der festen, hier beschriebenen Ablagestruktur kenntlich gemacht [7].

Die Ablagestruktur ist derzeit als Verzeichnisstruktur auf Dateisystemebene realisiert. In Abb. 5 werden die Hauptverzeichnisse der aktuellen Ablagestruktur des experimentspezifischen Datenbereichs der SFB501-Erfahrungsdatenbank dargestellt. Die einzelnen Verzeichnisse werden ausschließlich mit Großbuchstaben benannt und Umlaute werden ersetzt (also zum Beispiel DURCHFUEHRUNG statt Durchführung).

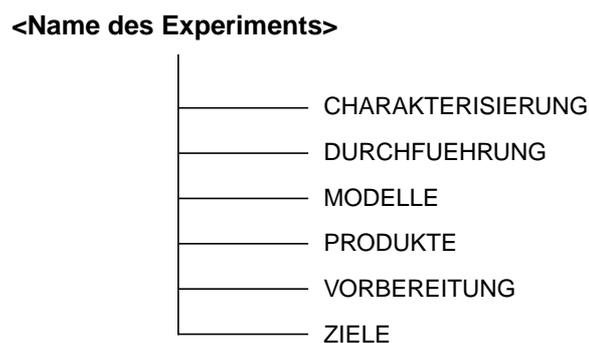


Abb. 5: Die Hauptverzeichnisse der Ablagestruktur

Eine Abbildung zwischen der Ablage- und der Zugriffstruktur ist durch die Angabe der jeweiligen Verzeichnisse in der Beschreibung der Zugriffstruktur bereits gegeben worden. Die dort verwendeten Abkürzungen für die Ablageverzeichnisse befindet sich in Anhang B.

Die Inhalte der Ablagestruktur sind größtenteils selbsterklärend. Des weiteren liefern die Erläuterungen zur Zugriffstruktur Angaben zu den Inhalten der einzelnen Unterverzeichnisse. Aus diesem Grund werden nur die wenigen Punkte erläutert, die nicht intuitiv erkennbar sind bzw. die sich im Inhalt von kontrollierten Experimenten unterscheiden.

6.2 Inhalte

In diesem Unterkapitel wird auf die Verfeinerung der Struktur, deren Inhalte sowie Unterschiede zwischen kontrollierten Experimenten und Fallstudien eingegangen.

6.2.1 CHARAKTERISIERUNG

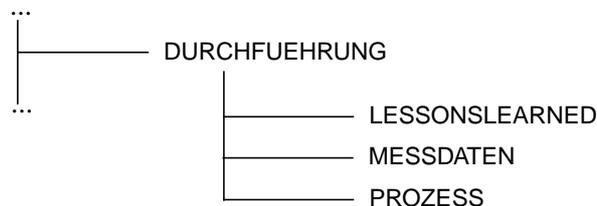
Unter CHARAKTERISIERUNG werden alle SE-Artefakte abgelegt, welche zur Charakterisierung des Experiments beitragen. Das Verzeichnis enthält keine weiteren (fixen) Unterverzeichnisse.

Beispiel

- *Kontext-Vektor* mit folgenden Angaben: Größe des Experiments (Personen, Personenmonate), Domäne, Umfeld (kommerziell, Forschung), zu berücksichtigende SE-Erfahrungen aus vergangenen Experimenten, Typ (Ersterstellung, Änderung/Ergänzung, ...)

6.2.2 DURCHFUEHRUNG

Der Strukturpunkt DURCHFUEHRUNG ist weiter verfeinert in:



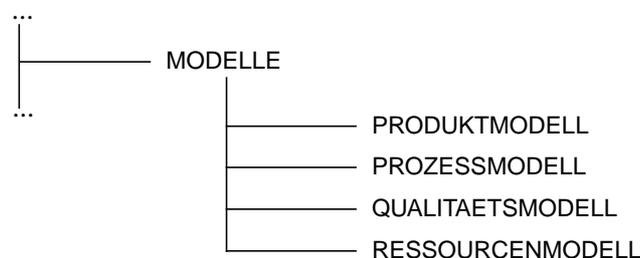
Neben den informell beschriebenen *Lessons learned* und den numerischen *Meßdaten* enthält dieses Verzeichnis jegliche Information, die den Ablauf des Entwicklungsprozesses beschreibt (Verzeichnis *Prozeß*)

Beispiel

- *Projekttrace* (z. B. Gegenüberstellung von Prozeßstatus und Zeit) im Falle der Fallstudie. Für kontrollierte Experimente würde hier die *experimentelle Prozedur* abgelegt werden.

6.2.3 MODELLE

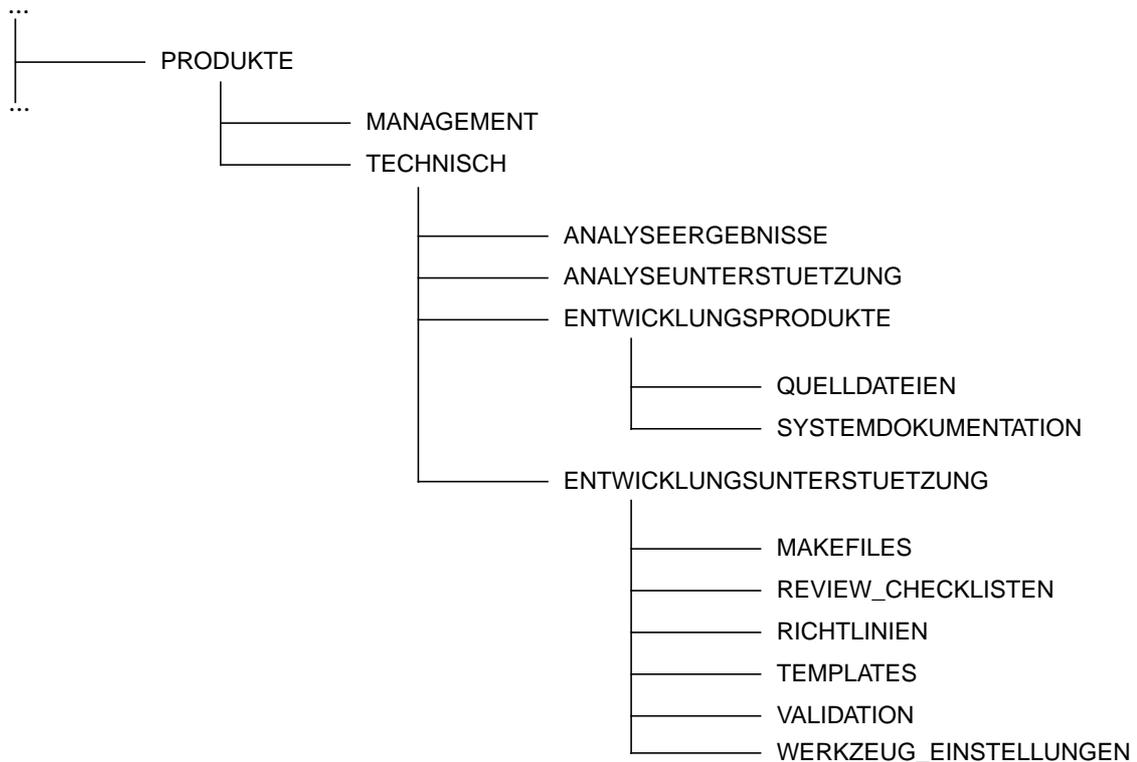
Der Strukturpunkt MODELLE ist weiter verfeinert in:



Planmodelle, die Basis für die Erstellung des Projektplans sind, werden in diesem Verzeichnis abgelegt. Zur Unterscheidung der verschiedenen Modelle gibt es entsprechende Unterverzeichnisse.

6.2.4 PRODUKTE

Der Strukturpunkt PRODUKTE ist weiter verfeinert in:



Dieser Bereich enthält alle SE-Artefakte, die als Instanz der oben beschriebenen Modelle gebildet werden können. Dies sind neben den in der Fallstudie anfallenden Entwicklungsprodukten auch Ergebnisse von Analysen sowie Unterstützungsprodukte für Entwicklung und Analyse.

Beispiele

- **MANAGEMENT:** *Fragebögen, Bildschirmmasken und Programme zur Erfassung von Meßdaten, Projektplan, Meßplan, Konfigurationsmanagement-Plan.* Für Experimente ist für dieses Verzeichnis der *experimentelle Entwurf* sowie der *experimentelle Meßplan* vorgesehen.
- **TECHNISCH:**
 - **ANALYSEERGERBNISSE:** *Folien von Zwischen- oder Endanalysen* der Meßdaten und deren Interpretation unterstützt durch *aufbereitete Lessons learned.*
 - **ANALYSEUNTERSTUETZUNG:** *Analysemodelle, Werkzeuge und Skripts zur Analyse.*
 - **ENTWICKLUNGSPRODUKTE:** *Quelldateien wie Kode oder StP-Daten.* Hierunter fallen alle Daten, die mit den entsprechenden Entwicklungswerkzeugen direkt gelesen werden können. Desweiteren befindet sich hier die *Dokumentation des zu entwickelnden Softwaresystems.*
 - **ENTWICKLUNGSUNTERSTUETZUNG:** *Makefiles* zur Generierung (z. B. des Softwaresystems oder einzelnen Komponenten), *Richtlinien* (z. B. *Programmierrichtlinien*), *Review-Checklisten*, *Templates*, *TAFs*. Für kontrollierte Experimente sind hier *Anleitungen* zur Durchführung des Experiments (z. B. in *REVIEW_CHECKLISTEN*) zu finden.

6.2.5 VORBEREITUNG

Der Strukturpunkt ZIELE enthält alle SE-Artefakte, welche zur Vorbereitung der Fallstudie angelegt werden.

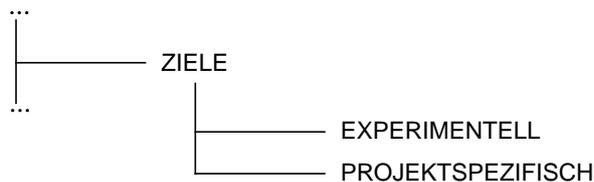
Der Strukturpunkt VORBEREITUNG enthält keine weiteren (fixen) Unterverzeichnisse.

Beispiele

- *Aufrufe* (z.B. e-mails oder Newsgroup-Artikel) oder Aushänge mit Ankündigungen eines Praktikums. *Check-Liste* für weitere Vorbereitungen (z. B. Einrichten von Accounts). *Leere Teilnehmerlisten*.

6.2.6 ZIELE

Der Strukturpunkt ZIELE ist weiter verfeinert in:



Experimentelle Ziele enthalten Angaben über die zu untersuchenden Eigenschaften von Techniken, Methoden und Werkzeugen, welche im Entwicklungsprozeß zum Einsatz kommen. Diese sollen in Bezug auf mögliche Verbesserungspotentiale hin untersucht werden. Einträge in diesem Bereich sind sowohl für Experimente als auch für Fallstudien sinnvoll.

Projektspezifische Ziele hingegen beziehen sich hauptsächlich auf Fallstudien. Sie beschreiben die Art der zu entwickelnden Software, seine Eigenschaften sowie organisatorische Einschränkungen bez. des Entwicklungsprozesses (z. B. welche Entwicklungssoftware oder -paradigma kommt zum Einsatz) als auch bez. des Managements (z. B. zeitliche oder finanzielle Rahmen)

Beispiel

- EXPERIMENTELL: *experimentelle Hypothesen, GQM's*
- PROJEKTSPEZIFISCH: *Projektbeschreibung, Ausschreibung*

6.3 Gesamtstruktur

<Name des Experiments>

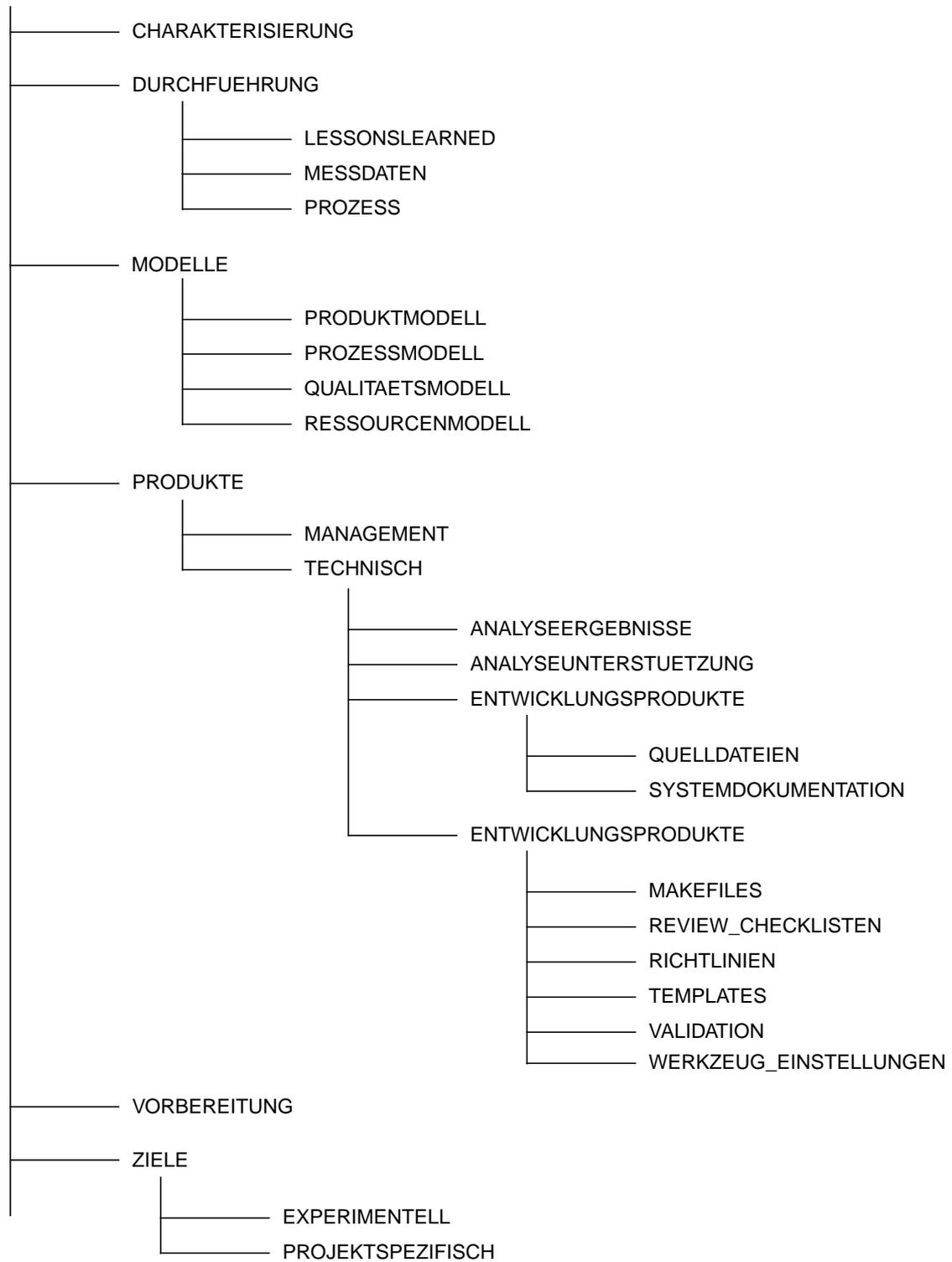


Abb. 6: Die Gesamtstruktur der Ablage

7 Automatisierung

7.1 Problembesprechung

Bei der Eingabe von Informationen, die in einen Datenbereich eingefügt werden sollen, müssen bestimmte Anforderungen erfüllt werden. So sollten z.B. die einzugliedernden Daten korrekt sein (z.B. sollten keine Rechtschreibfehler auftreten oder das Eingabeformat eines Datums sollte eingehalten werden). Die Erfüllung dieser Anforderung kann mit Hilfe einer genau festgelegten Eingabestruktur unterstützt werden. Eine Eingabestruktur besteht zum einen aus Eingabemasken (z.B. Textfelder, innerhalb derer textuelle Eingaben erfolgen können, oder Auswahllisten, aus denen vordefinierte Elemente ausgewählt werden können) und zum anderen aus Texten, die dem Benutzer hilfreiche Informationen bez. des Eingabevorganges anbieten.

Um den Eingabevorgang übersichtlicher zu gestalten, sollte die Eingabestruktur eine nach bestimmten Gesichtspunkten erfolgte Gliederung aufweisen, so daß dem Benutzer die Bedeutung der von ihm anzugebenden Informationen klarer wird. Ein Aspekt, nach dem gegliedert werden könnte, wäre die Optionalität der Angabe der Daten. So könnte die Eingabe von obligatorischen (d.h. unentbehrlichen, in jedem Fall anzugebenden) Daten vor der Eingabe der optionalen (d.h. nicht unbedingt vom Benutzer anzugebenden) Daten geschehen. Die Gliederung könnte auch einen Teil der logischen Struktur des Datenbereichs widerspiegeln. Werden z.B. aus den eingegebenen Daten mehrere Dokumente erzeugt, so macht es Sinn, die Eingabemasken von Daten, die zur Erzeugung eines speziellen Dokuments herangezogen werden, in der Eingabestruktur an benachbarten Orten anzusiedeln. Ein allgemeiner Vorteil einer Gliederung der Eingabestruktur ist die erleichterte Wartung und Erweiterbarkeit derselben.

Eine weitere Anforderung bei der Eingabe ist die Gewährleistung, daß alle angegebenen Informationen zueinander konsistent sind. Angenommen eine der angeforderten Informationen wäre der vollständige Name samt Anrede einer Person und eine weitere obligatorische Angabe wäre das Geschlecht derselben Person. Die Angabe "Frau Brunhilde Meyer" für den Namen und die Anrede wäre inkonsistent zur Geschlechtsangabe "männlich" (in der Mehrzahl der Fälle jedenfalls) und daher fehlerhaft. Zur Vermeidung solcher Inkonsistenzen sollten Daten, die zu anderen Daten in einer genau definierbaren Beziehung stehen, automatisch generiert werden. So könnte, in Anlehnung an das obige Beispiel, das Geschlecht einer Person anhand ihrer Anrede (Herr oder Frau), die der Benutzer eingibt, automatisch ermittelt werden, denn das Geschlecht einer Person steht in einer eindeutigen Beziehung zur Anrede dieser Person. Durch dieses Vorgehen reduziert sich zudem der Eingabeaufwand für den Benutzer, wodurch sich gleichsam der Bedienungskomfort erhöht.

Im obigen Abschnitt wurde vorgeschlagen, die Anzahl und den Umfang der vom Benutzer erwarteten Eingaben zu minimieren, indem bei der Dokumenterzeugung verwendete Informationen - wenn möglich - von bereits vorhandenen Informationen (z.T. durch vom Benutzer abgefragt) automatisch abgeleitet werden. Die Einführung von Automatisierung bei der Dokumenterzeugung und -ablage ist ebenfalls von Vorteil, wenn die Zugriffsstruktur auf die Dokumente und Informationen des Datenbereichs von der Ablagestruktur des Datenbereichs verschieden ist. Der Benutzer benötigt somit kein Wissen für die Ablage der erzeugten Dokumente, was den für die korrekte Durchführung der Eingabe erforderlichen Einarbeitungsaufwand vermindert. Dadurch sinkt wiederum die Fehlerwahrscheinlichkeit beim Eingabe-

vorgang, und es wird zudem gewährleistet, daß die Konventionen bez. der Ablage der Dokumente im Datenbereich eingehalten werden.

Die Umsetzung der in den obigen Abschnitten aufgeführten Überlegungen wird im nächsten Abschnitt besprochen.

7.2 Überblick über die praktische Umsetzung des Eingabevorgangs

Die EDB dient der Schematisierung und Archivierung von Experimenten verschiedenen Typs des Bereichs "Fallstudien" bzw. des Bereichs "Kontrollierte Experimente", d.h. der Eingabevorgang bezieht sich auf die Aufnahme neuer Experimente in die EDB. Das Eingabe-Skript wird über einen Link am Boden der Übersichtsseite (s. Abbildung 7) gestartet.

The screenshot shows a web page titled "Experiment-spezifischer Datenbereich: Fallstudien und kontrollierte Experimente". Below the title is a grey bar with the text "Übersichtsseite: Fallstudien und kontrollierte Experimente". A blue link "Zur SFB-EB Hauptseite" is centered below this bar. The main content is divided into two columns. The left column is headed "Fallstudien" and contains a section "SFB501 Team-Projekte" with two entries: "CoDEX" (Initial training experiment) and "Team1" (cross-project case study). The right column is headed "Kontrollierte Experimente" and contains two sections: "Lesetechniken" (comparing PBR and Ad-hoc reading) and "Fehlererkennungstechniken" (comparing abstraction-based, functional, and structural testing). Each entry includes links for "Übersichtsseite", "Ablagestruktur", and "Kontext-Vektor". At the bottom, a blue link "Neues Experiment in die Erfahrungsdatenbank aufnehmen" is displayed.

Abb. 7: Ausschnitt der Übersichtsseite "spezifisch.html"

Die Aufnahme eines neuen Experiments erfordert die Ergänzung bestehender HTML-Dokumente und zusätzlich die Erzeugung neuer HTML-Dokumente in der EDB (Zugriffstruktur). Zudem wird eine experiment-spezifische Verzeichnisstruktur (Ablagestruktur) angelegt.

Da die Navigationsdokumente (über die der Dokumentzugriff erfolgt) der EDB ausschließlich in HTML formuliert werden, wurde der Entschluß gefaßt, die Struktur und den Ablauf des Eingabevorgangs mit Hilfe eines HTML-Dokuments zu realisieren, welches mit einem Standard-Browser zur Ansicht gebracht werden kann. Dieses HTML-Dokument wird im weiteren Verlauf des Textes als "Eingabedokument" bezeichnet und realisiert die Eingabestruktur. Die leichte Erlernbarkeit von HTML ist ebenfalls für die unproblematische Wartung des Eingabedokuments äußerst nützlich.

Als Eingabemasken wurden Textfelder und Check-Boxen gewählt.

Die eigentliche Überprüfung der Eingaben und die Änderung und Erzeugung von Dokumenten sowie die Erschaffung der Verzeichnisstruktur eines Dokuments werden durch ein in PERL formuliertes CGI-Skript vollzogen, welches die benötigten Parameter durch das Eingabedokument übermittelt bekommt. Auf das CGI-Skript "insertESD" wird sich fortan mit der naheliegenden Bezeichnung "CGI-Skript" bezogen.

Im nächsten Kapitel wird der inhaltliche Aufbau des Eingabedokuments betrachtet.

7.3 Struktur des Eingabedokuments

Das Eingabedokument ist in sechs größere Abschnitte - absofort als Hauptabschnitte bezeichnet - unterteilt, wobei einige dieser Abschnitte in weitere Abschnitte - fortan als Unterabschnitte bezeichnet - aufgeteilt werden. An dieser Stelle wird nur ein kurzer schlagwortartiger Überblick über den inhaltlichen Aufbau des Eingabedokuments gegeben, auf den in den folgenden sechs Unterkapiteln (Kapitel 7.3.1 bis Kapitel 7.3.6) detailliert eingegangen wird.

- Hauptabschnitt A: Inhaltliche Gesamtübersicht (Kapitel 7.3.1)
- Hauptabschnitt B: Festlegung allgemeiner Eingabeparameter (Kapitel 7.3.2)
- Hauptabschnitt C: Festlegung der Eingabeparameter bez. des Kontextvektors (Kapitel 7.3.3)
- Hauptabschnitt D: Festlegung der Eingabeparameter bez. der Team-Mitglieder des Experiments (Kapitel 7.3.4)
- Hauptabschnitt E: Erzeugung und Einfügen der Dokumente in die EDB (Kapitel 7.3.5)
- Hauptabschnitt F: Hilfeabschnitt bez. der Eingabe deutscher Umlaute und des "ß" (Kapitel 7.3.6)

Um das Navigieren innerhalb des Dokuments zu erleichtern, kann von jedem Unterabschnitt aus durch dokument-interne Links zum vorangehenden (abgesehen vom ersten Unterabschnitt des Eingabedokuments) bzw. zum nachfolgenden Unterabschnitt (ausgenommen der letzte Unterabschnitt des Dokuments) gesprungen werden. In jedem ersten bzw. letzten Unterabschnitt eines Hauptabschnitts existiert außerdem ein Link zur inhaltlichen Dokumentübersicht.

Die oben angegebene inhaltliche Übersicht des Eingabedokuments wird nun im Detail erläutert.

7.3.1 Hauptabschnitt A: Inhaltliche Gesamtübersicht

Der erste Unterabschnitt beinhaltet eine inhaltliche Gesamtübersicht des Dokuments, über die die einzelnen Hauptabschnitte über dokument-interne Links anwählbar sind.

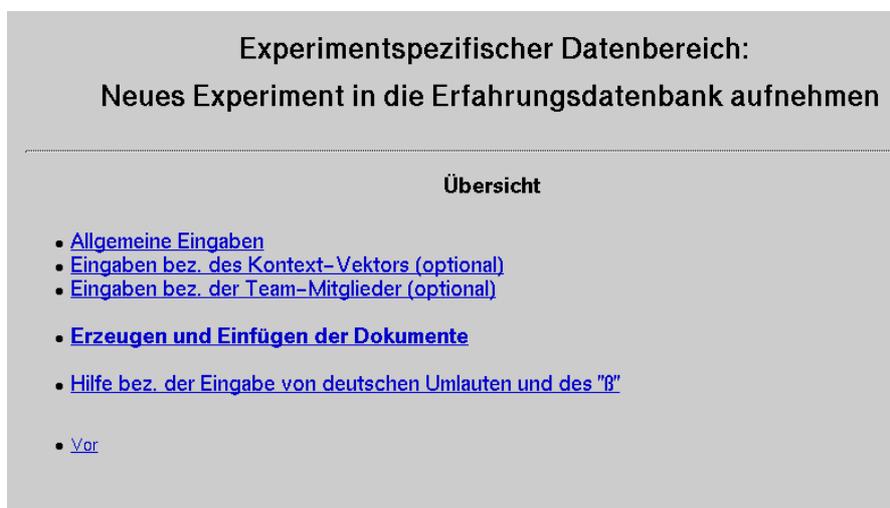


Abb. 8: Inhaltliche Gesamtübersicht des Eingabedokuments

7.3.2 Hauptabschnitt B: Festlegung allgemeiner Eingabeparameter

Innerhalb von Hauptabschnitt B werden allgemeine Eingabeparameter festgelegt, die u.a. zur Erzeugung der Hauptseite des Experiments herangezogen werden. Einige dieser Parameter sind obligatorisch anzugeben und entsprechend gekennzeichnet.

Hauptabschnitt B besteht aus sieben Unterabschnitten, wobei in jedem Unterabschnitt die Eingabe genau eines Parameters behandelt wird.

- Unterabschnitt B1: Inhaltliche Übersicht von Hauptabschnitt B (Kapitel 7.3.2.1)
- Unterabschnitt B2: Obligatorische Angabe des Experimenttyps (Kapitel 7.3.2.2)
- Unterabschnitt B3: Obligatorische Angabe des Experimenttitels (Kapitel 7.3.2.3)
- Unterabschnitt B4: Obligatorische Angabe des vollständigen Hauptpfadnamens des EDB-Verzeichnisses (Kapitel 7.3.2.4)
- Unterabschnitt B5: Obligatorische Angabe des Verzeichnisnamens des Experiments (Kapitel 7.3.2.5)
- Unterabschnitt B6: Optionale Angabe der Namen der Team-Mitglieder des Experiments (Kapitel 7.3.2.6)
- Unterabschnitt B7: Obligatorische Angabe einer kurzen Experimentbeschreibung (Kapitel 7.3.2.7)

In den folgenden sieben Kapiteln werden die Unterabschnitte von Hauptabschnitt B im einzelnen besprochen.

7.3.2.1 Unterabschnitt B1: Inhaltliche Übersicht von Hauptabschnitt B

Unterabschnitt B1 enthält eine inhaltliche Übersicht von Hauptabschnitt B, über die mit Hilfe von dokument-internen Links auf die sechs folgenden Unterabschnitte referenziert wird.



Abb. 9: Inhaltliche Übersicht von Hauptabschnitt B

7.3.2.2 Unterabschnitt B2: Obligatorische Angabe des Dokumenttyps

Im zweiten Unterabschnitt von Hauptabschnitt B kann einer von vier verschiedenen Experimenttypen aus einem Check-Box-Menü ausgewählt werden. Dabei existieren die folgenden, bisher unterschiedenen vier Experimenttypen: "Technische Fallstudien (SE-I-Typ)", "techn. & organ." Fallstudien (SE-II-Typ), Team-Projekte (CoDEx-Typ) sowie kontrollierte Experimente.

Defaultmäßig ist als Experimenttyp der "CoDEx-Typ" eingestellt.

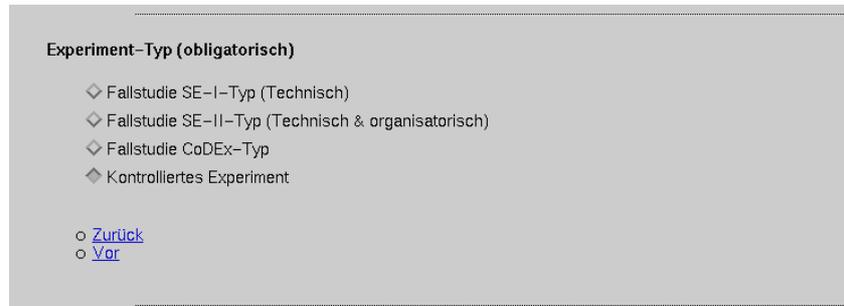


Abb. 10: Obligatorische Angabe des Experimenttyps

7.3.2.3 Unterabschnitt B3: Obligatorische Angabe des Experimenttitels

Im Unterabschnitt B3 wird der Titel des Experiments angegeben.

Für die Experimenttypen "SE-I-Typ" und "SE-II-Typ" werden Hinweise bez. der Wahl des Titels angezeigt (s. Abbildung 11). So hat bei Experimenten vom Typ "SE-I-Typ" bzw. "SE-II-Typ" der Titel wie folgt auszusehen: "SE-I-Praktikum <jahr_beginn>" bzw. "SE-II-Praktikum <jahr_beginn>/<jahr_ende>". Dabei ist <jahr_beginn> ein Platzhalter für die Jahreszahl des Jahres, in dem der Anfangszeitpunkt des Experiments liegt. <jahr_ende> ist ein Platzhalter für die letzten beiden Ziffern der Jahreszahl des Jahres, in dem der Endzeitpunkt des Experiments liegt.

Der Experimenttitel wird in sämtlichen von dem CGI-Skript modifizierten und erzeugten Dokumenten angezeigt.

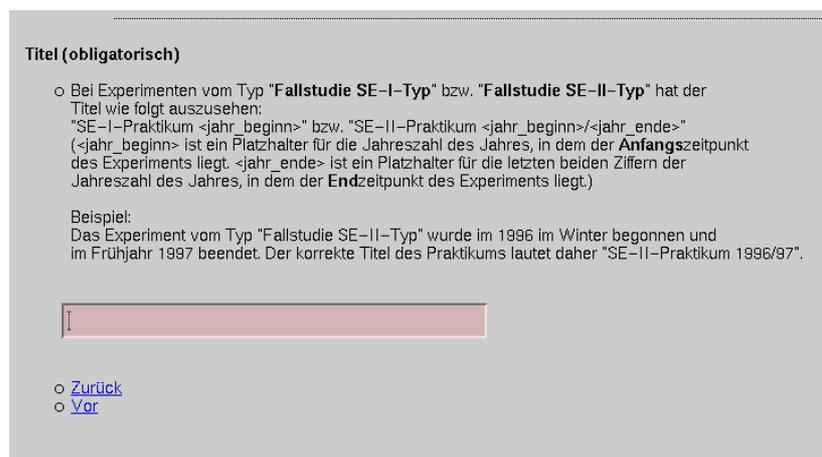


Abb. 11: Obligatorische Angabe des Experimenttitels

7.3.2.4 Unterabschnitt B4: Obligatorische Angabe des vollständigen Hauptpfadnamens des EDB-Verzeichnisses

Hier kann der Hauptpfad des EDB-Verzeichnisses festgelegt werden. Die angezeigte Voreinstellung dieses Parameters sollte jedoch nicht verändert werden, da eine Änderung nur dann von Relevanz ist, wenn das gesamte EDB-Verzeichnis an eine neue Position verschoben wird.

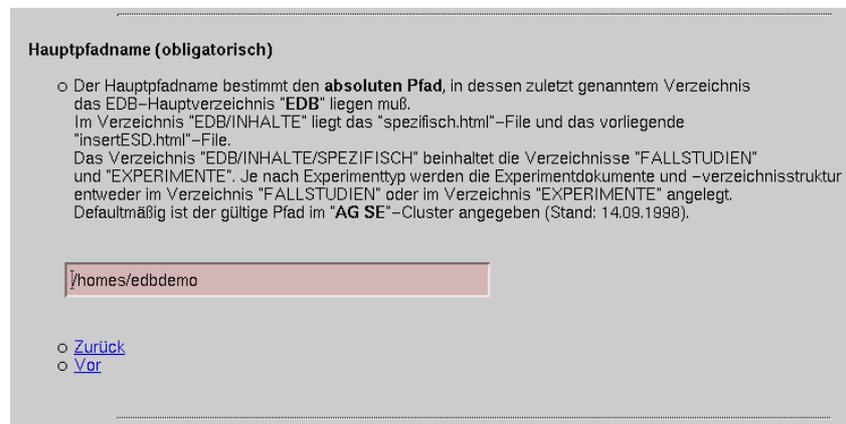


Abb. 12: Obligatorische Angabe des vollständigen Hauptpfadnamens des EDB-Verzeichnisses

7.3.2.5 Unterabschnitt B5: Obligatorische Angabe des Verzeichnisnamens des Experiments

Bei der Angabe des Verzeichnisnamens des Experiments sind bei den Experimenttypen "SE-I-Typ" und "SE-II-Typ" einige Konventionen zu beachten. Beim Experimenttyp "SE-I" hat der Verzeichnisname die Struktur "SE-1-<Anfangsjahr>"; beim Experimenttyp "SE-II" hat der Verzeichnisname die Struktur "SE-2-<Endjahr>". "<Anfangsjahr>" ist ein Platzhalter für die letzten beiden Ziffern des Jahres, das den Startzeitpunkt des Experiments vom "SE-I-Typ" kennzeichnet. "<Endjahr>" ist ein Platzhalter für die letzten beiden Ziffern des Jahres, das den Endzeitpunkt des Experiments vom "SE-II-Typ" bezeichnet. Die Angabe des Experimentverzeichnisnamens bei Experimenten vom "SE-II-Typ" unterscheidet sich somit von deren Struktur des Experimenttitels, bei dem Anfangs- **und** Endzeitpunkt des Experiments vermerkt werden (Kapitel 7.3.2.3).

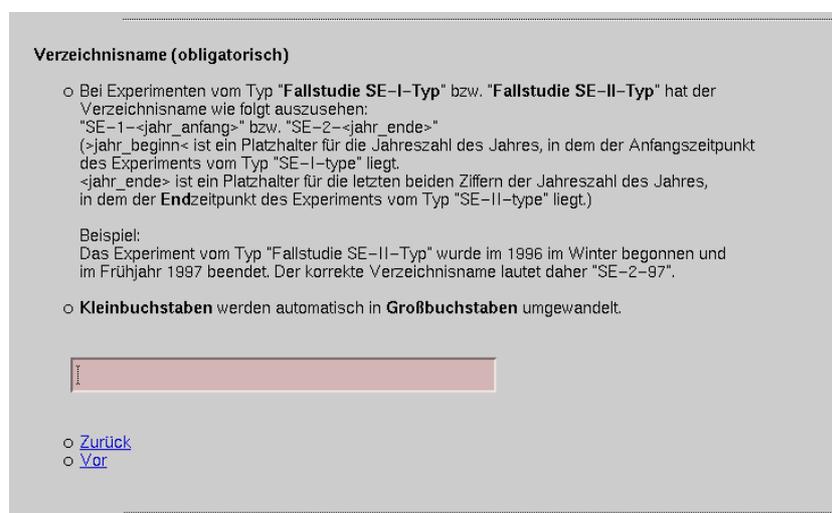


Abb. 13: Obligatorische Angabe des Verzeichnisnamens des Experiments

Das Experimentverzeichnis wird bei den Experimenttypen des Bereichs "Fallstudien" (das betrifft die Experimenttypen "SE-I-Typ", "SE-II-Typ" und "CoDEx-Typ") im Verzeichnis "EDB/INHALTE/SPEZIFISCH/FALLSTUDIEN", bei Experimenttypen des Bereichs "kontrollierte Experimente" im Verzeichnis "EDB/INHALTE/SPEZIFISCH/EXPERIMENTE" angelegt.

Durch das CGI-Skript werden Kleinbuchstaben automatisch in Großbuchstaben umgewandelt, damit die Konvention, Verzeichnisnamen ohne Kleinbuchstaben anzugeben, eingehalten wird (s. [7])

7.3.2.6 Unterabschnitt B6: Optionale Angabe der Namen der Teammitglieder des Experiments

In dem Textfeld von Unterabschnitt B6 können in alphabetischer Reihenfolge und durch Bindestriche voneinander getrennt die Namen aller Teammitglieder (Professoren, Assistenten, Studenten) eingegeben werden. Wird diese Angabe unterlassen, wird defaultmäßig durch das CGI-Skript "authors not yet available" als Parameterwert erzeugt.

Diese Angabe wird lediglich im Kopfteil der Experimenthauptseite dargestellt, und soll nur einen kurzen Überblick über die Namen der am Experiment beteiligten Personen verschaffen. Die im Hauptabschnitt D (Kapitel 7.3.4) eintragbaren, detaillierteren Informationen bez. der Teammitglieder werden auf der Teammitglieder-Seite des Experiments angezeigt.

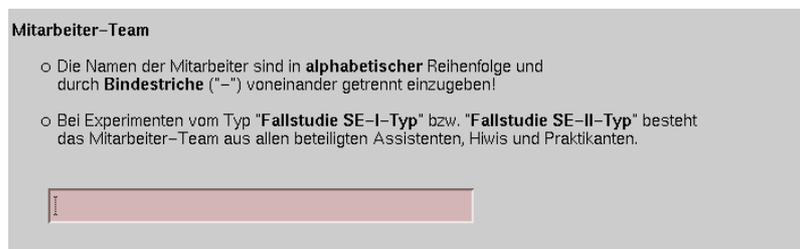


Abb. 14: Optionale Angabe der Namen der Team-Mitglieder des Experiments

7.3.2.7 Unterabschnitt B7: Obligatorische Angabe der Kurzbeschreibung des Experiments

Im Unterabschnitt B7 ist eine kurze Beschreibung des Experiments anzugeben, die von dem CGI-Skript auf der "spezifisch_de.html"¹-Seite im Verzeichnis "EDB/INHALTE" eingefügt wird. Über die "spezifisch_de.html"-Seite ist eine Zugriffsstruktur realisiert, über die auf die Hauptseiten und Kontextvektor-Seiten aller Experimente des "Fallstudien"- und des "Kontrollierte Experimente"-Bereichs zugegriffen werden kann (Abbildung 7).

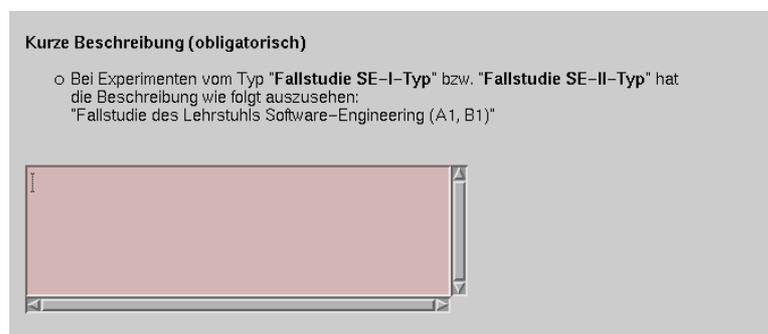


Abb. 15: Obligatorische Angabe einer kurzen Beschreibung des Experiments

1. bzw. "spezifisch.html für die englische Version

7.3.3 Hauptabschnitt C: Festlegung des Kontext-Vektors

Eingaben bez. des Kontext-Vektors

- [Zurück](#)
- [Vor](#)
- [Zurück zur Übersicht](#)

Bereich

Organisation

Erfahrung der Entwickler

Zahl der Entwickler

Anwendungsdomäne

Anforderungsbeschreibungs-Technik

Entwurfs-Technik

Komponenten-Typ

**Implementierungs-Technik/
Programmiersprache**

Inspektions-Technik

- [Zurück](#)
- [Vor](#)

**Prozessmodell/
Lebenszyklusmodell**

Fallstudien-Typ

Experiment-Typ

Entwicklungs-Richtlinien

Hauptabschnitt C besteht aus einem einzigen Unterabschnitt. Innerhalb dieses Unterabschnitts können die folgenden, allesamt optionalen Parameter bez. der Einordnung des Experiments (s. Bereich, ...) und des Kontext-Vektors (s. Erfahrung, ...) [14] festgelegt werden. Zum Zeitpunkt der Fertigstellung dieses Dokuments können die folgenden Parameter festgelegt werden [15] :

- Bereich: Defaultmäßig "Experiment-specific"
- Organisation: Defaultmäßig "University of Kaiserslautern"
- Erfahrung der Entwickler
- Zahl der Entwickler
- Anwendungsdomäne
- Anforderungsbeschreibungs-Technik
- Entwurfs-Technik
- Komponenten-typ
- Implementierungs-Technik/ Programmiersprache
- Inspektions-Technik
- Prozeßmodell/ Lebenszyklusmodell
- Projekt-Typ
- Experiment-Typ
- Entwicklungs-Richtlinien

Bemerkung:

Folgende Einträge des Kontext-Vektors werden aus anderen Parametern generiert und sind vom Benutzer nicht anzugeben (für diese Angaben existieren also keine Eingabemasken im Eingabedokument).

- Name: entspricht dem in Unterabschnitt B3 (s. Kapitel 7.3.2.3) von Hauptabschnitt B angegebenen Experimenttitel
- Bereich: wird anhand des in Unterabschnitt B2 (s. Kapitel 7.3.2.2) von Hauptabschnitt B gewählten Experimenttyps automatisch generiert. Mögliche Werte: {Fallstudien | Kontrollierte Experimente}
- Typ: entspricht dem in Unterabschnitt B2 (s. Kapitel 7.3.2.2) von Hauptabschnitt B gewählten Experimenttyp

Abb. 16: Hauptabschnitt C

7.3.4 Hauptabschnitt D: Festlegung der Eingabeparameter bez. der Team-Mitglieder des Experiments

Im Hauptabschnitt D können u.a. die Namen und Initialen der am Experiment beteiligten Professoren, Assistenten sowie Studenten eingetragen werden. Die angegebenen Informationen sind dann nach Generierung aller Experimentdokumente (Kapitel 7.3.5) auf der Team members-Seite des Experiments einsehbar. Die in Unterabschnitt B6 (Kapitel 7.3.2.6) von Hauptabschnitt B eingetragenen Namen der am Experiment beteiligten Personen werden dahingegen nur im Kopfteil der Experimenthauptseite - in alphabetischer Reihenfolge - angezeigt.

Hauptabschnitt D besteht aus fünf Unterabschnitten.

- Unterabschnitt D1: Inhaltliche Übersicht von Hauptabschnitt D (Kapitel 7.3.4.1)
- Unterabschnitt D2: Wichtiger Hinweis bez. einzuhaltender Konventionen (Kapitel 7.3.4.2)
- Unterabschnitt D3: Angabe der Namen der am Experiment beteiligten Professoren (Kapitel 7.3.4.3)
- Unterabschnitt D4: Angabe der Namen der am Experiment beteiligten Assistenten (Kapitel 7.3.4.4)
- Unterabschnitt D5: Angabe der Namen der am Experiment beteiligten Studenten (Kapitel 7.3.4.5)

7.3.4.1 Unterabschnitt D1: Inhaltliche Übersicht von Hauptabschnitt D

Unterabschnitt D1 besteht aus einer inhaltlichen Übersicht von Hauptabschnitt D, über die die nachfolgenden vier Unterabschnitte referenziert werden.



Abb. 17: Inhaltliche Übersicht von Hauptabschnitt D

7.3.4.2 Unterabschnitt D2: Wichtiger Hinweis bez. einzuhaltender Konventionen

Im Unterabschnitt D2 wird ein wichtiger Hinweis bez. des Formats der in den Unterabschnitten D3, D4 und D5 zu machenden Eingaben angezeigt. Da die Namen mit den Initialen in einer ungeordneten Liste auf der Team members-Seite erscheinen, muß vor bzw. nach jedem Eintrag der Daten einer Person ein "" bzw. ein "" eingetragen werden, damit die Daten zu jeder Person als selbständiger Listeneintrag dargestellt werden. Soll ein Student mit Namen "Sledge Coder", den Initialen "SC" und dem Status "Praktikant" eingetragen werden, ist innerhalb der zugehörigen Eingabemaske (Kapitel 7.3.4.5) die Zeichenfolge "Sledge Coder (SC) (Praktikant)" einzutragen. Die student-spezifischen Angaben sind auf die Weise als Listeneintrag gekennzeichnet. Das CGI-Skript erledigt diese Formatierung derzeit noch nicht selbständig.

Es ist dabei zu beachten, daß die vergebenen Initialen eindeutig sein müssen. Gibte es bspw. zwei Studierende namens Martina Spieß und Michael Schmidt, so wären die Initialen MS1 und MS2 oder MSP und MSc denkbar.

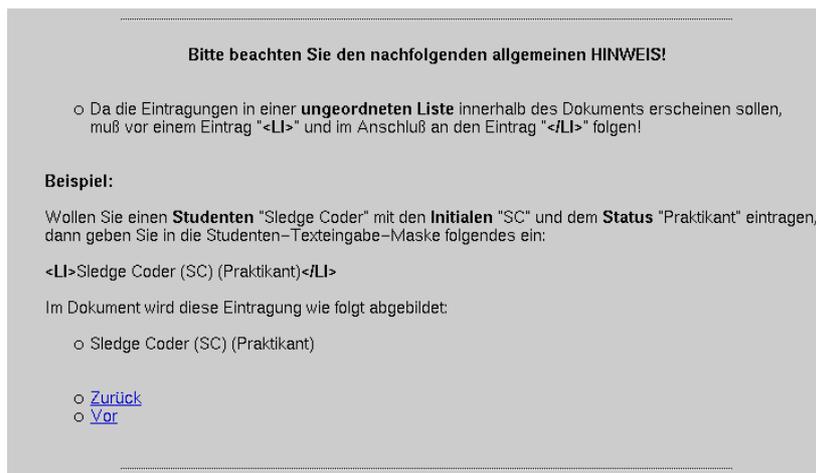


Abb. 18: Wichtiger Hinweis bez. einzuhaltender Konventionen

7.3.4.3 Unterabschnitt D3: Angabe der Namen der am Experiment beteiligten Professoren

Im Unterabschnitt D3 sind die Namen und Initialen (in Klammern gesetzt) der am Experiment beteiligten Professoren untereinander einzutragen.

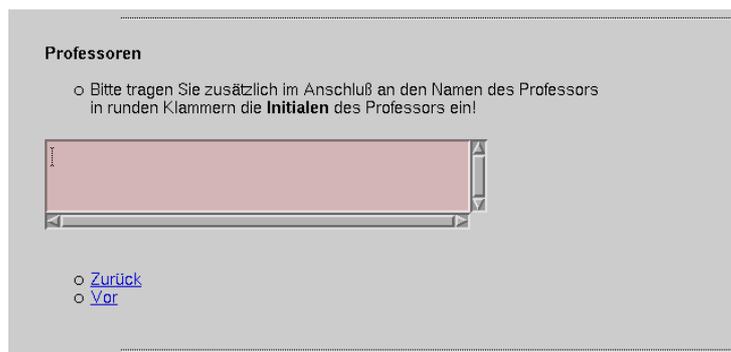


Abb. 19: Angabe der Namen der am Experiment beteiligten Professoren

7.3.4.4 Unterabschnitt D4: Angabe der Namen der am Experiment beteiligten Assistenten

Im Unterabschnitt D4 können die Namen und Initialen (in Klammern gesetzt) der am Experiment beteiligten Assistenten eingetragen werden.

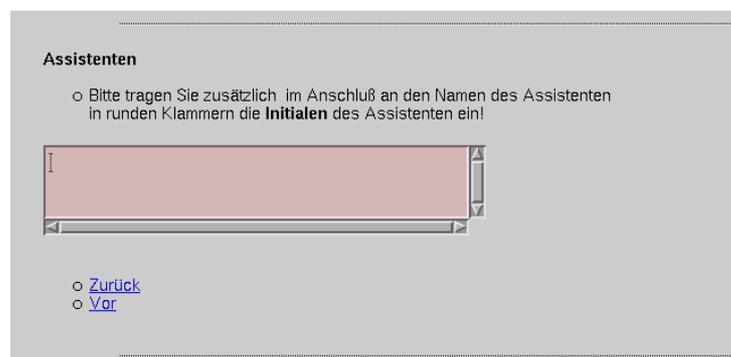


Abb. 20: Angabe der Namen der am Experiment beteiligten Assistenten

7.3.4.5 Unterabschnitt D5: Angabe der Namen der am Experiment beteiligten Studenten

Im Unterabschnitt D5 sind die Namen der Studenten zusammen mit den zugehörigen Initialen sowie mit dem Status (HiWi oder Praktikant) des Studenten anzugeben. Initialen und Status sind jeweils in Klammern gesetzt einzugeben.

Studenten

- Bitte tragen Sie zusätzlich im Anschluß an den Namen des Studenten, jeweils in runden Klammern, die **Initialen** sowie den **Status** (HiWi oder Praktikant) des Studenten ein!
- Beim Experimenttyp "Fallstudie **SE-I-Typ**" sollen nur die Namen der Studenten des **besten** Teams eingetragen werden, da nur die Daten dieser Gruppe für die Aufnahme in die EDB aufbereitet werden!

- [Zurück](#)
- [Vor](#)
- [Zurück zur Übersicht](#)

Abb. 21: Angabe der Namen der am Experiment beteiligten Studenten

7.3.5 Hauptabschnitt E: Erzeugen und Einfügen der Dokumente in die EDB

Hauptabschnitt E enthält ein Button ("Erzeugen und Einfügen"), dessen Betätigung eine Abfolge von Aktionen auslöst:

Zuerst werden die im Hauptabschnitt B (Kapitel 7.3.2), C (Kapitel 7.3.3) und D (Kapitel 7.3.4) eingetragenen Parameterwerte an das CGI-Skript "insertESD" übersendet.

Das CGI-Skript kontrolliert nun, ob der Benutzer für alle obligatorischen Parameter eine gültige Angabe vornahm. Im Fehlerfall wird eine Fehlermeldung in Form eines HTML-Dokuments an den Browser zurückgeschickt.

Danach wird eine experiment-spezifische Verzeichnisstruktur [7] angelegt, in der die Dokumente des Experiments abgelegt werden. Über die "spezifisch_de.html"-Seite im Verzeichnis "EDB/INHALTE" ist dann ein direkter Zugriff (Ablagestruktur) auf die Verzeichnisstruktur des Experiments möglich.

Im nächsten Schritt wird/werden die Hauptseite(n) des Experiments erzeugt und in das Experimenthauptverzeichnis eingefügt. Auf die Experimenthauptseite kann ebenfalls wie auf die Experimentverzeichnisstruktur von der "spezifisch.html"-Seite im Verzeichnis "EDB/INHALTE" aus zugegriffen werden.

Abschließend wird die Teammitglieder-Seite erzeugt und in das Unterverzeichnis "CHARAKTERISIERUNG" des Experimenthauptverzeichnisses eingefügt sowie die KontextVektor-Seite generiert, die in das Verzeichnis "CONTEXT_VECTOR" des Bereichs "Fallstudien" (bei Experimenten vom Typ "SE-I", "SE-II" oder "CoDEx") bzw. "kontrollierte Experimente" abgelegt wird.

Genauere Angaben bez. der Arbeit des CGI-Skripts befinden sich in Anhang D.



Abb. 22: Erzeugen und Einfügen der Dokumente

7.3.6 Hauptabschnitt F: Hilfeabschnitt bez. der Eingabe deutscher Umlaute und des "ß"

Im Hauptabschnitt F werden Hilfestellungen bez. der Eingabe von deutschen Umlauten und dem deutschen "ß" gegeben.

Dabei werden verschiedene Methoden zur Erzeugung der oben angegebenen Zeichen erläutert.

Bei der ersten Methode wird - je nach Tastaturtyp - ein Umlaut durch Niederdrücken entweder der META-, der COMPOSE- oder der ALT GRAPH-Taste zusammen mit der Taste des dem Umlaut zugrundeliegenden Vokals erzeugt. Zusätzliches Niederhalten der SHIFT-Taste erzeugt den zugehörigen Großbuchstaben.

Bei der zweiten Methode gibt der Benutzer einen den Umlaut repräsentierenden HTML-Code an.

"ä" bzw. "Ä" entspricht "ä" bzw. "Ä".

"ö" bzw. "Ö" entspricht "ö" bzw. "Ö".

"ü" bzw. "Ü" entspricht "ü" bzw. "Ü".

"ß" entspricht "ß".

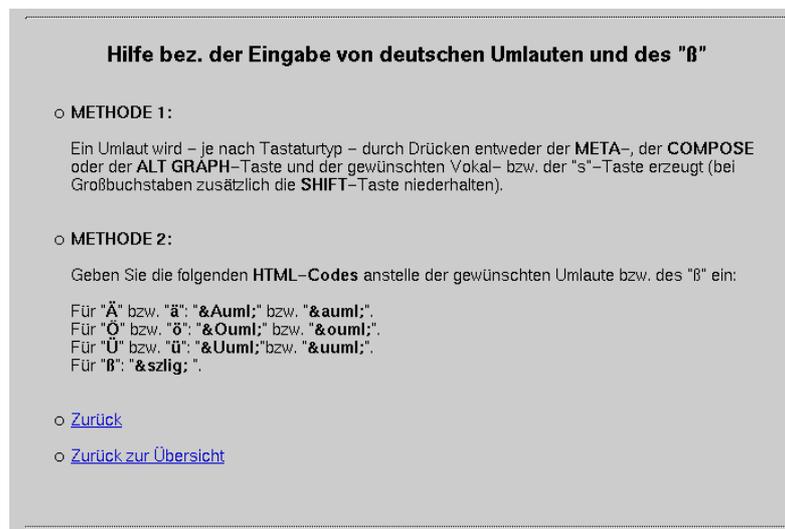


Abb. 23: Hilfeabschnitt bez. der Eingabe deutscher Umlaute und des "ß"

8 Zusammenfassung und Ausblick

8.1 Zusammenfassung

Dieser Bericht beschreibt die Zugriffs- und Ablagestruktur eines Teils der Erfahrungsdatenbank des SFB 501: Fallstudien aus dem experimentspezifischen Datenbereich. Beide Strukturen werden strukturell und inhaltlich beschrieben. Die Zugriffsstruktur beschreibt dabei eine daten-/produktorientierte Variante des QIP, die Ablagestruktur ist nach Produkttypen gegliedert. Ebenfalls dokumentiert ist die Abbildung zwischen diesen beiden Strukturen. Dies soll den manuellen Einlagerungsvorgang unterstützen. Wir haben auch gezeigt, daß eine weitere Unterscheidung zwischen verschiedenen Typen von Fallstudien notwendig ist. Wir sind auf die Evaluierung der Strukturen eingegangen und haben begründet, wieso diese ein vorübergehend fixes Ergebnis darstellt. Um eine initiale Erzeugung beider Strukturen zu unterstützen, wurde ein Generierungssystem implementiert, welches hierin ausführlich dokumentiert ist. Außerdem ist für typische, zu erwartende Änderungen in den Strukturen beschrieben, an welchen Stellen im Generierungssystem welche Änderungen wie vorzunehmen sind.

8.2 Ausblick

Die Arbeit im experimentspezifischen Bereich der Erfahrungsdatenbank ist mit diesem Bericht natürlich keineswegs abgeschlossen. Wir können uns in vielen Themenbereichen Verbesserungen und Erweiterung vorstellen, die im folgenden kurz angedacht sind.

8.2.1 Sichtenbasierte Zugriffe

Trotz Strukturierung bereitet der Zugriff über die beschriebenen Zugriffsstruktur vielen internen und externen "Beobachtern" eines Experiments Schwierigkeiten. Nur wenige wollen wirklich auf jede Information zugreifen, den meisten reicht ein Ausschnitt. So interessiert sich ein Entwickler bspw. für die Projektunterstützung des QIP-Schrittes drei und den Entwicklungsbereich aus QIP-Schritt vier.

Vorstellbar ist eine rollenbasierte Sichtendefinition, die die Informationsflut auf die für entsprechende Rolle relevante Informationen filtert. In einem ersten Schritt könnten fixe Sichten, z.B. die des Planers, Manders und Entwicklers, zur Verfügung gestellt werden. In einem weiteren Schritt sind frei definierbare Sichten vorstellbar. Dies muß mit der aktuellen Portierung des übergreifenden Teils der EDB auf ein objekt-relacionales DB-System abgeglichen werden.

8.2.2 Schnittstelle für die Einlagerung

Wir mußten feststellen, daß die Komplexität der aktuellen Strukturen eine maßgebliche Rolle bei der fehlerhaften Einlagerung von SE-Erfahrungen spielt. Zusätzlich ist —z. B. gerade in der Durchführungsphase, in der eine relativ große Anzahl von Personen (Manager und Entwickler) Ergebnisse produzieren— die Aufgabe der für die Einlagerung verantwortlichen Rolle mit einem hohen Aufwand verbunden. Folglich wäre es schön, wenn ein Teil der Arbeit verteilt werden könnte, so daß z. B. auch Entwickler

ihre Ergebnisse einfach einlagern könnten. Jedoch ist es den Entwicklern weder zuzumuten noch bringen sie die nötige Motivation auf, sich neben ihrer Entwicklungstätigkeit in das QIP einzuarbeiten. Folglich kann den einzelnen Personen die Verantwortung zur Einlagerung der individuellen Ergebnisse nicht übertragen werden.

Für die Durchführungsphase ist eine Kopplung des Produktmanagementsystems an den experimentenspezifischen Datenbereich denkbar. Ist eine Version eines Produktes abgeschlossen, so könnte bspw. beim Eincheck-Vorgang ein automatisiertes Update in der Erfahrungsdatenbank (Ablagestruktur) durchgeführt werden.

8.2.3 Evolution der Struktur

Bisher steht für den Bereich der kontrollierten Experimente nur eine rudimentäre Zugriffstruktur zur Verfügung. Für die momentan eingelagerten kontrollierten Experimente ist sie jedoch ausreichend. Sollten in Zukunft verstärkt kontrollierte Experimente durchgeführt und eingelagert werden, so ist sicherlich eine Überarbeitung und Evaluierung dieser Struktur notwendig. Gleiches, wenn auch in geringerem Umfang, gilt auch für den Bereich der Fallstudien.

8.2.4 Generierung

Die derzeit vorliegende Version des Eingabedokuments und des CGI-Skripts sind in mehreren Punkten noch verbesserungswürdig, die in den nachfolgenden Kapitel angeführt werden.

8.2.4.1 Beseitigung des Bugs bei der Übersendung von Fehlermeldungen seitens des CGI-Skripts

Das CGI-Skript überprüft zu Beginn seiner Ausführung die Vollständigkeit der Angabe der obligatorischen Parameter. Wurde für einen obligatorischen Parameter durch den Benutzer kein Wert angegeben, wird eine Fehlermeldung generiert und an den Browser in Form eines HTML-Dokuments geschickt. Im Normalfall sollte die Fehlermeldung als HTML-Seite angezeigt werden, doch in zahlreichen Fällen wird entweder der HTML-Code als Textdokument durch den Browser angezeigt anstatt interpretiert zu werden, oder der Browser meldet, daß das Fehlermeldungsdocument leer ist. Da zur Erzeugung und Übersendung der Fehlermeldungsdocumente für jeden vom CGI-Skript abgefangenen Fehler das gleiche Codegerüst verwendet wird, bleibt dieses Fehlverhalten vorläufig ungeklärt.

8.2.4.2 Einführung von Auswahllisten

Für bestimmte Parameter soll eine Auswahlliste aufrufbar sein, über die standardisierte Werte ausgewählt werden können, so daß zum einem für den Benutzer ein geringerer Eingabeaufwand anfällt und zum anderen die Möglichkeit eines Eingabefehlers reduziert wird. So könnten z.B. vor allem bei Eingaben bez. des Kontextvektors Auswahllisten realisiert werden, da viele Kontextvektorparameter nur Parameterwerte einer genau festgelegte Wertemenge als Parameterwerte erwarten. Diese festgelegte Wertemenge könnte bequem und ohne des Risikos eventueller Schreibfehler aus einer Auswahlliste ausgesucht werden. Allerdings sollte immer die Möglichkeit bestehen, eine Alternative zu den Werten der Auswahlliste angeben zu können, damit Änderungen innerhalb der Parameterwertemengen nicht unmittelbar auch eine Änderung des Eingabedokuments erforderlich machen.

8.2.4.3 Automatisch alphabetische Anordnung der Namen der Team-Mitglieder eines Experiments

Die in den Unterabschnitten D3 (Kapitel 7.3.4.3), D4 (Kapitel 7.3.4.4) und D5 (Kapitel 7.3.4.5) von Hauptabschnitt D eingegebenen Namen der Team-Mitglieder sollten automatisch in eine alphabetische Reihenfolge gebracht. Dadurch wird die Auffindbarkeit eines speziellen Namens eines Team-Mitglieds in einer Namensliste (z.B. auf der Experimenthauptseite oder der Team members-Seite) erhöht.

8.3 Danksagung

Hiermit möchten wir allen Personen, die an der initialen Entwicklung sowie der fortlaufenden Anpassung der beschriebenen Strukturen an gegebene Situationen beteiligt waren, danken. Dies sind —neben den in der Autorenliste Erwähnten— insbesondere

- Antje von Knethen, die während der Durchführung der CoDEx-Fallstudie unermüdlich berechtigte Änderungen für die damals initiale Struktur, besonders der Zugriffstruktur, einbrachte,
- Raimund L. Feldmann und Jürgen Münch, die beide hauptsächlich an der Entwicklung der initialen Struktur mitwirkten und auf eine gemeinsame Ablagestruktur für Fallstudien und kontrollierte Experimente hinarbeiteten, und
- Marcus Trapp, der in der Endphase dieser Version des Berichts hochmotiviert Feedback in einer Review-Sitzung gab.

Anhang A

Literatur

- [1] Sonderforschungsbereich 501. Entwicklung großer Systeme mit generischen Methoden. Finanzierungsantrag 1995-1996-1997. Universität Kaiserslautern. 1994
- [2] Victor R. Basili and Richard W. Selby and David H. Hutchens. Experimentation in Software Engineering. Transactions in Software Engineering. Jul. 1986, vol. se-12. pages 733--743.
- [3] Barbara Ann Kitchenham. Evaluating software engineering methods and tools, part 1: The evaluation context and evaluation methods. Software Engineering Notes, Jan. 1996, vol. 21, pages 11-15.
- [4] Victor R. Basili. *The Experience Factory and its relationship to other improvement paradigms*. In Ian Sommerville and Manfred Paul, editors, Proceedings of the Fourth European Software Engineering Conference, pages 68–83. Lecture Notes in Computer Science Nr. 717, Springer-Verlag, 1993.
- [5] Victor R. Basili and H. Dieter Rombach. The TAME Project: Towards Improvement-oriented Software Environments. IEEE Transactions on Software Engineering, SE-14(6):758–773, June 1988
- [6] Raimund L. Feldmann, Stefan Vorwieger. The web-based Interface to the SFB 501 Experience Base. Technical Report No. 01/98, Sonderforschungsbereich 501, Dept. of Computer Science, University of Kaiserslautern, 67653 Kaiserslautern, Germany, 1998.
- [7] Marion Fechtig. Bewertung der Zugriffs- und Ablagestruktur für Fallstudien der SFB-Erfahrungsdatenbank mittels Integration von SE-Praktika. Projektarbeit, Fachbereich Informatik, Universität Kaiserslautern, 67653 Kaiserslautern, January 1998
- [8] Lothar Baum, Barbara Dellen, Erik Kamsties, Antje von Knethen, Stefan Vorwieger. Modeling Real-Time Systems with SCR - An Evaluation and Lessons Learned in a Building Automation System Project. Technical Report 7/97. Sonderforschungsbereich 501, Dept. of Computer Science, University of Kaiserslautern. 67653 Kaiserslautern, Germany, 1997.
- [9] Gotzhein (Leiter), Nehmer (Leiter), Dellen, Kollnischko, Kronenburg, Molter, Münch, Peper, Queins, Rothkugel, Verlage, Zeckzer. Entwicklung einer Kontrollsystemarchitektur. Ergebnisbericht Team2. Universität Kaiserslautern, 1997.
- [10] Stefan Vorwieger, Antje von Knethen, Bin Shen, "CoDEX: allgemeine Experimentbeschreibung", SFB-Bericht 9/1997. Universität Kaiserslautern, 1997.
- [11] Raimund L. Feldmann and Birgit Geppert and Frank Rößler. First Results from an Experimental Evaluation of SDL-Pattern based Protocol Design. Technical Report 3/99. Sonderforschungsbereich 501, Dept. of Computer Science, University of Kaiserslautern. 67653 Kaiserslautern, Ger-

- many, 1999.
- [12] Raimund L. Feldmann, Jürgen Münch, Stefan Queins, Stefan Vorwieger, Gerhard Zimmermann. Baselineing a Domain-Specific Software Development Process. Technical Report 02/1999. Sonderforschungsbereich 501, Dept. of Computer Science, University of Kaiserslautern. 67653 Kaiserslautern, Germany, 1999
 - [13] Natalie Ardet, Raimund L. Feldmann, Antje von Knethen, Jürgen Münch, Stefan Vorwieger. Softwaresystemdokumentation eines Entwicklungsprojekts zur Erstellung eines eingebetteten Gebäudeautomationssystems mit UML. Entwicklungsprojekt, Fachbereich Informatik, Universität Kaiserslautern, 67653 Kaiserslautern, 1999.
 - [14] Raimund L. Feldmann, Jürgen Münch, Stefan Vorwieger. Towards Goal-Oriented Organizational Learning: Representing and Maintaining Knowledge in an Experience Base. In Proceedings of the Tenth International Conference on Software Engineering and Knowledge Engineering (SEKE'98), pages 236–245, San Francisco Bay, CA, USA, June 1998.
 - [15] Raimund L. Feldmann, Wolfgang Mahnke, Norbert Ritter. (OR)DBMS-Support for the SFB 501 Experience Base. Technical Report 12/1998. Sonderforschungsbereich 501, Dept. of Computer Science, University of Kaiserslautern. 67653 Kaiserslautern, Germany, 1998

Anhang B

Abkürzungen der Ablagestruktur

Abk.	Verzeichnis				
CV	../CONTEXT_VECTOR				
C	CHARAKTERISIERUNG				
DL	DURCH- FUEHRUNG/ (D)	LESSONSLEARNED			
DM		MESSDATEN			
DP		PROZESS			
EU	../.../EDB/UEBERGREIFEND/...				
MPD	MODELLE/ (M)	PRODUKTMODELL			
MPZ		PROZESSMODELL			
MQ		QUALITAETSMODELL			
MR		RESSOURCENMODELL			
PM	PRODUKTE/ (P)	MANAGEMENT			
PTAE		TECHNISCH/ (PT)	ANALYSEERGERBNISSE		
PTAU			ANALYSEUNTERSTUETZUNG		
PTEPQ			ENTWICKLUNGS- PRODUKTE/ (PTEP)	QUELLDATEIEN	
PTEPS				SYSTEMDOKUMENTATION	
PTEUM			ENTWICKLUNGS- UNTERSTUETZUNG/ (PTEU)	MAKEFILES	
PTEURC				REVIEW_CHECKLISTEN	
PTEURI				RICHTLINIEN	
PTEUT				TEMPLATES	
PTEUV				VALIDATION	
PTEUW				WERKZEUG_EINSTELLUNGEN	
V				VORBEREITUNG	
ZE			ZIELE/ (Z)	EXPERIMENTELL	
ZP				PROJEKTSPEZIFISCH	

Anhang C

Vorgehen bei Änderungen des Generierungs-Systems

In diesem Kapitel wird für geänderte Anforderungen an die Eingabestruktur das Vorgehen bei der Modifikation des Eingabedokuments "insertESD_de.html"¹ im Verzeichnis "EDB/INHALTE" sowie des CGI-Skriptes "insertESD" im Verzeichnis "~httpdaem/htbin/edbdemo_htbin" erläutert.

Die folgenden Modifikationsvorhaben werden behandelt.

- Hinzufügen eines neuen Experimenttyps (Kapitel C.1)
- Hinzufügen eines neuen Attributs des Kontextvektors (Kapitel C.2)
- Ändern der Experimentverzeichnisstruktur (Kapitel C.3)

C.1 Hinzufügen eines neuen Experimenttyps

Beim Hinzufügen eines neuen Experimenttyps müssen insgesamt drei Dateien modifiziert werden, und zwar die Datei "spezifisch_de.html"² (Übersichtsseite des experiment-spezifischen Bereichs) im Verzeichnis "EDB/INHALTE", das Eingabedokument "insertESD.html" im Verzeichnis "EDB/INHALTE" und das CGI-Skript "insertESD" im Verzeichnis "~httpdaem/htbin/edbdemo_htbin".

C.1.1 Modifikationen im Dokument "spezifisch_de.html"

Für den neuen Experimenttyp muß wie für jeden bereits schon bestehenden Experimenttyp eine neue Sektion angelegt werden, welche zur Aufnahme der Verweise auf die aufzunehmenden Dokumente des Experimenttyps dient. Innerhalb des diese Sektion beschreibenden HTML-Codes ist die Position, vor der der HTML-Code für einen neu hinzuzufügenden Experimenteintrag durch das CGI-Skript eingefügt werden soll, durch ein eindeutiges, als Kommentar markiertes Schlüsselwort zu kennzeichnen.

1. bzw. der Datei "insertESD.html" für die englische Variante

2. bzw. die Datei "spezifisch.html" für die englische Variante

C.1.2 Modifikationen im Eingabedokument

Im Unterabschnitt B2 des Hauptabschnitts B des Eingabedokuments, in dem der Experimenttyp eines neu hinzuzufügenden Experiments festgelegt wird, muß eine weitere Eingabemaske vom Typ "Radio Button" ergänzt werden. Diese Eingabemaske hat den folgenden Aufbau.

```
<INPUT type=radio name="doku" value="<Dokumenttyp-Name>"> Experimenttyp-Name
```

<Experimenttyp-Name> ist ein Platzhalter für die Bezeichnung des Experimenttyps, die im Kontextvektor für Eintrag "Fallstudien-Typ" bzw. "Experiment-Typ" verwendet wird. Dieser Wert wird ebenfalls im CGI-Skript bei jeder Fallunterscheidung bez. der verschiedenen Experimenttypen verwendet.

C.1.3 Modifikationen im CGI-Skript

Mit Hilfe der ausführlichen Kommentare sollte sich der Benutzer in den Aufbau des CGI-Skriptes einarbeiten und nach genauer Durchsicht die zu ändernden Stellen erkennen und modifizieren können. Welche Abschnitte im CGI-Skript speziell zu ändern sind, hängt maßgeblich vom Experimenttyp ab, so daß allgemeine Aussagen im voraus nur sehr begrenzt gemacht werden können.

Prinzipiell müssen sämtliche Fallunterscheidungen, die den Experimenttyp betreffen, begutachtet werden. Das betrifft z.B. den Abschnitt, in dem die Verzeichnisstruktur des Experiments angelegt wird (Kapitel C.3) und bestimmte Parameterzuweisungen, die das CGI-Skript automatisch vornimmt (bei der Durchsicht des CGI-Skripts zu ermitteln).

Ebenfalls ist der Abschnitt im CGI-Skript zu erweitern, in dem auf der "spezifisch_de.html"-Seite ein Eintrag (enthält den Experimenttitel und die Experimentkurzbeschreibung sowie Links auf die Experimenthauptseite, die Kontextvektor-Seite und die Experimentverzeichnisstruktur) für ein Experiment des neu hinzuzufügenden Experimenttyps eingefügt wird. An dieser Stelle wird im "spezifisch_de.html"-Dokument nach einem experimenttyp-spezifischen Schlüsselwort (als HTML-Kommentar im "spezifisch_de.html"-Dokument integriert) gesucht, vor dessen Position durch das CGI-Skript der HTML-Code für den neu hinzuzufügenden Experimenteintrag ergänzt wird.

Weiterhin muß der HTML-Code der zu erzeugenden Experimenthauptseite in dem Abschnitt integriert werden, in dem auch die Experimenthauptseiten der bisher bekannten vier Experimenttypen erzeugt werden. Dabei ist zu beachten, daß alle vom Skript kreierten und modifizierten Dokumente RCS-verwaltet werden.

Abschließend müssen noch die Kommentare an die aktuellen Modifikationen des CGI-Skripts angeglichen werden.

C.2 Hinzufügen eines neuen Attributs des Kontextvektors

Falls dem Kontextvektor ein neues Attribut hinzugefügt werden soll, sind insgesamt zwei Dokumente zu bearbeiten. Zum einen muß das Eingabedokument "insertESD.html" im Verzeichnis "EDB/INHALTE" modifiziert werden, desweitern noch das CGI-Skript "insertESD" im Verzeichnis "~httpdaem/htbin/edbdemo_htbin".

C.2.1 Modifikation des Eingabedokuments

Im Eingabedokument ist der Hauptabschnitt C, in dem die Parameterwerte bez. des Kontextvektors angegeben werden können, anzupassen.

Die hinzuzufügende Eingabemaske wird durch den folgenden HTML-Code realisiert.

```
<font face=HELVETICA size=+1><B><Attribut-Name></B>
```

`<INPUT TYPE=TEXT NAME="<Attribut-Parameter-Name>" SIZE=40>`

`<Attribut-Name>` ist ein Platzhalter für die Bezeichnung des Attributs, wie sie auch im Kontextvektor benutzt wird.

`<Attribut-Parameter-Name>` ist ein Platzhalter für den Parameternamen, mit Hilfe dessen das CGI-Skript den eingetragenen Wert abfragen kann.

C.2.2 Modifikation des CGI-Skriptes

Im CGI-Skript ist lediglich der siebte Abschnitt (Anhang D), in dem die Kontextvektor-Seite erzeugt wird, zu ergänzen.

Zuerst muß an das Skript der (evtl. durch den Benutzer angegebene) Parameterwert des neuen Attributs übergeben werden. Das geschieht durch den folgenden Code, der zu Beginn des siebten Abschnitts einzufügen ist.

```
if (defined($in{'<Attribut-Parameter-Name' }) and ($in{'<Attribut-Parameter-Name' } ne "")) {
  $<Attribut-Parameter-Name = $in{'<Attribut-Parameter-Name' };
} else {
  $<Attribut-Parameter-Name = "";
}
```

`<Attribut-Parameter-Name>` ist ein Platzhalter für den Parameternamen, mit Hilfe dessen das CGI-Skript den eingetragenen Wert abfragen kann.

Als nächstes muß der HTML-Code, der die Kontextvektor-Seite beschreibt, erweitert werden, damit erstens der Name des neuen Attributs und zweitens der evtl. durch den Benutzer festgelegte Wert (im Hauptabschnitt C des Eingabedokuments angegeben; Kapitel 7.3.3) angezeigt werden kann.

Der folgende HTML-Code (vom HTML-Code der Tabelle durch Fettschrift abgehoben) ist in der Kontextvektor-Tabelle an der gewünschten Stelle einzutragen.

Hier zunächst der die Tabelle definierende HTML-Code (nicht einzutragen!).

```
<table border BGCOLOR="#FFFFFF">
<TR BGCOLOR= #CCCCCC>
<TH><FONT FACE="HELVETICA" SIZE=4>Factor</FONT></TH>
<TH><FONT FACE="HELVETICA" SIZE=4>Value</FONT></TH>
</TR>
```

...

Dieses in Fettschrift dargestellte HTML-Code-Segment ist zu ergänzen.

```
<tr>
  <td width=200><font face=HELVETICA size=+1><Attribut-Name></font></td>
  <td width=200><font face=HELVETICA size=+1>${<Attribut-Parameter-Name></font></td>
</tr>
```

...

Der HTML-Code, der die Tabellendefinition beendet (nicht einzugeben!).

```
</table>
```

<Attribut-Name> ist ein Platzhalter für die Bezeichnung des Attributs, wie sie auch im Kontextvektor benutzt wird.

<Attribut-Parameter-Name> ist ein Platzhalter für den Parameternamen, mit Hilfe dessen das CGI-Skript den eingetragenen Wert abfragen kann.

Damit sind alle am CGI-Skript durchzuführenden Modifikationen abgeschlossen.

C.3 Ändern der Experimentverzeichnisstruktur

Zum Ändern der Experimentverzeichnisstruktur eines bereits eingefügten Experimenttyps braucht nur das CGI-Skript "insertESD" im Verzeichnis "~httpdaem/htbin/edbdemo_htbin" modifiziert zu werden.

C.3.1 Modifikation des CGI-Skripts

Im CGI-Skript sind die nötigen Änderungen im zweiten Abschnitt vorzunehmen. Dort wird die Experimentverzeichnisstruktur angelegt. Die angelegte Verzeichnisbasisstruktur [7], die allen Experimenttypen gemein ist, wird durch den nachfolgenden PERL-Code erzeugt.

```
if(chdir $dir){
umask 000;
if (!mkdir ($vname, 0777)) {
print "<H3><BLINK>Error:</BLINK> Creation of the directory '$dir/$vname' failed!</H3><BR>\n";
print "<HR>\n</BODY>\n</HTML>\n";
exit 0;
} else {
print "<H3>The directory '$dir/$vname' has been created successfully!</H3>\n";
if(chdir $vname){

system "mkdir Uebergreifend";
system "mkdir CHARAKTERISIERUNG";
system "mkdir DURCHFUEHRUNG";
system "mkdir MODELLE";
system "mkdir PRODUKTE";
system "mkdir ZIELE";
system "mkdir VORBEREITUNG";

chdir "CHARAKTERISIERUNG";
system "mkdir RCS";
chdir ".";

chdir "DURCHFUEHRUNG";
system "mkdir LESSONSLEARNED";
system "mkdir MESSDATEN";
system "mkdir PROZESS";

chdir "../MODELLE";
system "mkdir PRODUKTMODELL";
system "mkdir PROZESSMODELL";
system "mkdir QUALITAETSMODELL";
```

```
system "mkdir RESSOURCENMODELL";

chdir "../PRODUKTE";
system "mkdir MANAGEMENT";
system "mkdir TECHNISCH";

chdir "MANAGEMENT";

chdir "../TECHNISCH";
system "mkdir ANALYSEERGEBNISSE";
system "mkdir ANALYSEUNTERSTUETZUNG";
system "mkdir ENTWICKLUNGSPRODUKTE";
system "mkdir ENTWICKLUNGSUNTERSTUETZUNG";

if (($doku =~ "se-i-type") || ($doku =~ "se-ii-type")) {
chdir "ENTWICKLUNGSPRODUKTE";

system "mkdir Quelldateien";
system "mkdir Systemdokumentation";

chdir "Quelldateien";
system "mkdir Kode";
system "mkdir Stp_omt";
chdir "..";

chdir "../ENTWICKLUNGSUNTERSTUETZUNG";

system "mkdir Makefiles";
system "mkdir Review_Checklisten";
system "mkdir Tests";
system "mkdir Werkzeug_Einstellungen";

chdir "..";
}

chdir "../ZIELE";
system "mkdir EXPERIMENTELL";
system "mkdir PROJEKTSPEZIFISCH";
chdir "..";
chdir "../ZIELE";
system "mkdir EXPERIMENTELL";
system "mkdir PROJEKTSPEZIFISCH";
chdir "..";

print "All corresponding directories have been created successfully!<P>\n";
chdir "..";
}
}
}
```

Welche Modifikationen in diesem Code-Segment vorzunehmen sind, ist experimenttyp-abhängig.

Daher seien abschließend nur die beiden relevanten Befehle zur Verzechniserzeugung und zum Verzeichniswechsel genannt.

Mit dem Befehl **'system "mkdir <Verzeichnisname>"** wird das Verzeichnis "<Verzeichnisname>" im aktuell eingestellten Verzeichnis angelegt.

Mit dem Befehl `'chdir "<Pfadname/Verzeichnisname>"'` wird als aktuelles Verzeichnis das Verzeichnis `"<Pfadname/Verzeichnisname>"` gewählt.

Anhang D

Dokumentation des Generierungs- Systems

Dieses Kapitel beinhaltet Auszüge der Kommentierung des CGI-Skriptes, die in den wesentlichen Aufbau und die Funktionsweise des Skriptes einführen.

Die folgenden Auszüge werden im folgenden angegeben.

- Skript-Erläuterung (Kapitel D.1)
- Skript-Beschreibung (Kapitel D.2)
- Das Skript aufrufende Files (Kapitel D.3)
- Vom Skript modifizierte Files (Kapitel D.4)
- Vom Skript erzeugte Files (Kapitel D.5)
- Aufbau des Skripts (Kapitel D.6)
- Obligatorische und optionale Skript-Parameter (Kapitel D.7)

D.1 Skript-Erläuterung

SKRIPT-ERLÄUTERUNG

- Skript-Name: "insertESD"
- Skript-Verzeichnis: "/homes/httpdaem/htbin/edbdemo_htbin/"
- Skript-Sprache: PERL

D.2 Skript-Beschreibung

Dieses PERL-Script legt eine neue Experimentverzeichnisstruktur sowie spezifische

HTML-Experimentdokumente im Verzeichnis

`$main_path/EDB/INHALTE/SPEZIFISCH/ {EXPERIMENTE | FALLSTUDIEN}` an.

"\$main_path" ist eine PERL-Variable, die den Wert eines gleichnamigen Parameters, der im "insertESD.html"-File festgelegt werden kann, beinhaltet.

BEMERKUNG:

Bei einer Änderung des obigen Pfadnamens brauchen nur die Pfadangaben angepaßt zu werden, die nach der Markierung "@Pfad" stehen.

Folgendes Vorgehen bei einer Pfadnamensänderung empfiehlt sich also:

Suche alle Markierungen "@Pfad" innerhalb dieses Files und ändere - wie beabsichtigt - die Pfadangaben, die nach den gefundenen Markierungen stehen.

Grundlage für die Erzeugung und das Einfügen der entsprechenden Dokumente ist das File "`$main_path/EDB/INHALTE/insertESD.html`".

Die erzeugten Dokumente sind über eine in die HTML-Übersichtsseite

`$main_path/EDB/INHALTE/spezifisch.html` eingetragene Linkstruktur zugänglich.

D.3 Das Skript aufrufende Files

Folgende Files verwenden das CGI-Skript.

- "`$main_path/EDB/INHALTE/insertESD.html`"

D.4 Vom Skript modifizierte Files

Folgende Files werden vom Skript modifiziert.

- "`$main_path/EDB/INHALTE/spezifisch.html`"

D.5 Vom Skript erzeugte Files

Für folgende Experiment-Typen können die Verzeichnisstruktur sowie spezifische Dateien angelegt werden.

----- Bereich "Fallstudien" -----

===== SE-I-type =====

Angelegte Dateien:

FALLSTUDIEN/se-1-<endjahr>_contents.html

FALLSTUDIEN/SE-1-<endjahr>/CHARAKTERISIERUNG/team.html

FALLSTUDIEN/CONTEXT_VECTOR/se-1-<endjahr>_contents.html

===== SE-II-type =====

Angelegte Dateien:

FALLSTUDIEN/se-2-<endjahr>_contents.html

FALLSTUDIEN/se-2-<endjahr>_contents_subprojekt.html

FALLSTUDIEN/SE-2-<endjahr>/CHARAKTERISIERUNG/team.html

FALLSTUDIEN/CONTEXT_VECTOR/se-2-<endjahr>_contents.html

===== CoDEx-type =====

FALLSTUDIEN/<Experimentverzeichnisname in Kleinbuchstaben>_contents.html

FALLSTUDIEN/<Experimentverzeichnisname>/CHARAKTERISIERUNG/team.html

FALLSTUDIEN/CONTEXT_VECTOR/<Experimentverzeichnisname in Kleinbuchstaben>_contents.html

----- Bereich "Kontrollierte Experimente" -----

===== Controlled experiment-type =====

Angelegte Dateien:

EXPERIMENTE/<Experimentverzeichnisname in Kleinbuchstaben>_contents.html

EXPERIMENTE/<Experimentverzeichnisname>/CHARAKTERISIERUNG/team.html

EXPERIMENTE/CONTEXT_VECTOR/<Experimentverzeichnisname in Kleinbuchstaben>_contents.html

BEMERKUNG:

Die Hauptseiten (bzw. die Subprojekt-Seiten) und die Kontextvektor-Dokumente werden RCS-verwaltet!

D.6 Aufbau des Skriptes

Folgende Aktionen werden bei Ausführung des CGI-Skriptes ausgeführt.

1. Testen, ob die obligatorischen Angaben vollständig sind, und Übernahme der Werte in PERL-Variablen (Bei Unvollständigkeit wird eine detaillierte Fehlermeldung ausgegeben).
2. Erzeugung der Verzeichnisstruktur des Experiments.
3. Öffnen des "spezifisch.html"-Files zum Lesen und Erzeugen eines ".temp.html"-Files, welches vorübergehend das modifizierte "spezifisch.html"-File enthält (siehe 8.).
4. Eintragen des Experiment-Links in das "spezifisch.html"-File.

5. Erzeugen der Experimenthauptseite im Verzeichnis {FALLSTUDIEN | EXPERIMENTE}

5(ii). Bei einem Experiment vom Typ "SE-II-type": Erzeugen der Subprojekt-Seite im Verzeichnis "FALLSTUDIEN".

6. Erzeugen des "team.html"-Files im Verzeichnis

"{FALLSTUDIEN | EXPERIMENTE}/<Experimentverzeichnisname>/CHARAKTERISIERUNG".

7. Erzeugen des Kontextvektors im Verzeichnis "{FALLSTUDIEN | EXPERIMENTE}/CONTEXT_VECTOR".

8. Auschecken des alten "spezifisch.html"-Files; Umbenennen des ".temp.html"-Files in "spezifisch.html" und Einchecken des neuen "spezifisch.html"-Files sowie Löschen des ".temp.html"-Files.

9. Verlassen des CGI-Skriptes

D.7 Obligatorische und optionale Skript-Parameter

Im CGI-Skript werden die nachfolgenden Parameter verwendet.

===== Obligatorische Parameter =====

- 'doku' (Experiment-Typ: {SE-I-type | SE-II-type | CoDEX-type | Controlled experiment-type})
- 'title' (Experimentname)
- 'main_path' (Hauptpfadname ohne '/' am Ende:
 - Default = "/homes/edbdemo" (in "insertESD.html" defaultmäßig gesetzt).
 - Im zuletzt angegebenen Verzeichnis des Pfades muß sich das EDB-Hauptverzeichnis "EDB" befinden.
 - Im Verzeichnis "EDB/INHALTE" befindet sich das "spezifisch.html"-File und das "insertESD.html"-File.
 - Im Verzeichnis "EDB/INHALTE/SPEZIFISCH" befinden sich die Verzeichnisse "FALLSTUDIEN" und "EXPERIMENTE". Je nach Experimenttyp werden die Dokumente mit zugehöriger Verzeichnisstruktur entweder im Verzeichnis "EXPERIMENTE" oder im Verzeichnis "FALLSTUDIEN" angelegt.)
- 'vname' (Verzeichnisname: alle Kleinbuchstaben werden in Großbuchstaben umgewandelt)
- 'note' (Experimentbeschreibung)

===== Optionale Parameter =====

- 'authors' (Studenten/Praktikanten/HiWis und evtl. betreuende Assistenten)

===== Optionale Parameter bzgl. des Kontextvektors =====

- 'cv_section' (Section: Default= "Experiment-specific")
- 'cv_org' (Organization: Default= "University of Kaiserslautern")
- 'cv_exp_of_dev' (Experience of developers)
- 'cv_number_of_dev' (Number of developers)
- 'cv_application_domain' (Application domain)

- 'cv_req_tech' (Requirements technique)
- 'cv_design_tech' (Design technique)
- 'cv_comp_type' (Component type)
- 'cv_impl_tech' (Implementation technique/ Programming language)
- 'cv_inspection_tech' (Inspection technique)
- 'cv_process_model' (Process model/ Life cycle model)
- 'cv_project_type' (Project type)
- 'cv_experiment_type' (Experiment type)
- 'cv_dev_guidelines' (Development guidelines)

BEMERKUNG:

Folgende Einträge des Kontextvektors werden aus anderen Parametern gewonnen bzw. generiert:

- 'Name' (entspricht 'title')
- 'Area' (generiert: {Case studies | Controlled experiments})
- 'Type' (entspricht 'doku')

===== Optionale Parameter bzgl. der Team-Mitglieder-Seite =====

- 'team_professoren' (Namen und Initialen der Professoren; Default= "")
- 'team_assistenten' (Namen und Initialen der Assistenten; Default= "")
- 'team_studenten' (Namen und Initialen der Studenten; Default= "")

BEMERKUNG:

Einem Eintrag für die Professoren-/ Assistenten-/ Studentennamensliste muß jeweils ein "" voran- und ein "" nachgestellt werden, da die Namen in eine ungeordnete Liste eingefügt werden sollen.

