

Rainer Ammermann

**Zielgruppengerechte
Aufbereitung und Kommunikation von
Geschäftsprozessmodellen als Teil des
betrieblichen Informationsmanagements**

Transformation von
UML-Prozessmodellen (Unified Modeling Language)
mittels XML-Technologie (eXtensible Markup Language)
für verschiedenartige Medien

Hausarbeit zur Diplomprüfung
an der
Hochschule für Angewandte Wissenschaften Hamburg
Fachbereich Bibliothek und Information
Studiengang Bibliotheks- und Informationsmanagement

vorgelegt von
Rainer Ammermann

Hamburg, Juli 2003

Referent: Prof. Dr. Martin Gennis

Korreferent: Dr. Lars Hagge

Zusammenfassung:

Diese Arbeit beschreibt, wie Geschäftsprozessmodelle einer Organisation effizient dokumentiert werden können, so dass sie für Management, Mitarbeiter und Kunden transparent sind und eine konkrete Unterstützung im betrieblichen Alltag darstellen.

Wesentliche Grundlagen der Geschäftsprozessmodellierung werden vorgestellt und ihre Bedeutung anhand eines Fallbeispiels veranschaulicht. Es werden spezifische Modellierungskonventionen erarbeitet, die eine bestimmte Anwendung der UML bei Geschäftsprozessmodellen definieren. Darauf aufbauend werden wesentliche Funktionen eines Publikationssystems entwickelt, das mit Hilfe der XML-Technologie Geschäftsprozessmodelle sowohl in interaktiv navigierbarer als auch in linearer, ausdrückbarer Form publiziert.

Inhaltsverzeichnis

1	Einleitung	13
1.1	Kontext	13
1.2	Ziele und Ergebnisse der Arbeit.....	15
1.3	Vorgehensweise und Gliederung.....	18
2	Grundlagen der Geschäftsprozessmodellierung	21
2.1	Management von Geschäftsprozessen	21
2.1.1	<i>Definition des Geschäftsprozesses</i>	21
2.1.2	<i>Geschäftsprozesse und Workflows</i>	22
2.1.3	<i>Aufgaben des Geschäftsprozessmanagements</i>	23
2.1.4	<i>Die Rolle des betrieblichen Informationsmanagements</i>	25
2.2	Verwendung von Geschäftsprozessmodellen	27
2.2.1	<i>Strategisches Management</i>	28
2.2.2	<i>Qualitätsmanagement</i>	29
2.2.3	<i>IT-Management</i>	30
2.2.4	<i>Mitarbeiter</i>	31
2.2.5	<i>Kunden und Geschäftspartner</i>	32
2.2.6	<i>Übersicht der Nutzergruppen</i>	32
2.3	Modellierung von Geschäftsprozessen.....	33
2.3.1	<i>Inhalt und Darstellung von Geschäftsprozessmodellen</i>	33
2.3.2	<i>Bestimmung von Modellsichten</i>	34
2.4	Die Unified Modeling Language(UML).....	35
2.4.1	<i>Grundlagen der UML</i>	35
2.4.2	<i>Darstellung der Funktionssicht</i>	37
2.4.3	<i>Darstellung der Organisationssicht (Prozess-Beteiligte)</i>	38
2.4.4	<i>Darstellung der Objektsicht (Hilfsmittel und Produkte)</i>	38
2.4.5	<i>Darstellung von Beziehungen und Strukturen</i>	39
2.4.6	<i>Darstellung der Prozesssicht auf Geschäftsprozessebene</i>	40
2.4.7	<i>Darstellung der Prozesssicht auf Workflowebene</i>	42
3	Fallbeispiel: Prozessdokumentation für den DESY- Anwendersupport.....	45
3.1	Beschreibung des Projektumfangs.....	45
3.1.1	<i>Anlass und Hintergrund des Projekts</i>	45
3.1.2	<i>Projektziele und -produkte</i>	47
3.1.3	<i>Angestrebter Nutzen</i>	47
3.1.4	<i>Vorgehensweise bei der Modellierung</i>	48
3.1.5	<i>Entwicklung eines Dokumentationsverfahrens</i>	48

3.2	Aufgaben und Prozesse des Anwendersupport.....	49
3.2.1	<i>Rolle des Anwendersupports innerhalb von DESY</i>	49
3.2.2	<i>Prozess „Arbeitsplatz einrichten“</i>	51
3.2.3	<i>Prozess „Anwender schulen“</i>	52
3.2.4	<i>Prozess „Anwender unterstützen“</i>	53
3.2.5	<i>Prozess „Dokumentation bereitstellen“</i>	54
3.2.6	<i>Prozess „System pflegen“</i>	56
3.2.7	<i>Zentrales Arbeitsmittel: Der Request-Tracker</i>	58
3.3	Anforderungen der Prozessdokumentation	59
4	Entwicklung spezifischer Modellierungskonventionen.....	61
4.1	Aufbau und Gliederung eines Modells.....	62
4.2	Funktionssicht	64
4.2.1	<i>Darstellung der Aufgabenhierarchie</i>	64
4.2.2	<i>Mehrfachverwendung von Aufgaben in verschiedenen Kontexten</i>	65
4.3	Organisationssicht	67
4.3.1	<i>Darstellung der Akteure</i>	67
4.3.2	<i>Darstellung von Tätigkeitsprofilen</i>	67
4.4	Objektsicht	69
4.4.1	<i>Darstellung von Objekten</i>	69
4.4.2	<i>Modellierung von Systemoberflächen</i>	70
4.5	Prozesssicht: Geschäftsprozessebene	72
4.6	Prozesssicht: Workflow-Ebene.....	73
4.6.1	<i>Darstellung von Handlungsanweisungen</i>	73
4.6.2	<i>Formale Definition der Nachrichten</i>	74
4.6.3	<i>Inhalt und Gliederung der Nachrichten</i>	75
4.6.4	<i>Bedienung von Informationssystemen</i>	77
4.7	Weitere Anforderungen der Prozessdokumentation.....	79
4.7.1	<i>Beschreibungstexte</i>	79
4.7.2	<i>Steuerung der Print-Dokumentation</i>	80
4.7.3	<i>Einbindung von Dokumenten</i>	81
5	Die Anwendersicht des Publikationssystems.....	83
5.1	Anforderungen und Kriterien der Anwendersicht.....	83
5.1.1	<i>Verwendung und Nutzen des Systems</i>	83
5.1.2	<i>Kriterien der Gebrauchstauglichkeit</i>	85
5.2	Aufbau der Anwendersicht	86
5.2.1	<i>Gestaltung der Systemoberfläche</i>	86
5.2.2	<i>Gestaltung der Printfassung</i>	91

5.3 Anwenderszenarien	93
5.3.1 Szenario 1: Bewältigung eines Vorgangs im Prozess „Anwender schulen“	93
5.3.2 Szenario 2 : Optimierung des Prozesses „System pflegen“	97
5.3.3 Szenario 3: Anfrage im Prozess „Arbeitsplatz einrichten“	101
6 Datensicht und Systemsicht.....	103
6.1 Datensicht	104
6.1.1 XML als Speicherformat der Modelldaten	104
6.1.2 Bewertung des Standards XMI	106
6.1.3 Entwicklung einer proprietären XML-Lösung	108
6.1.4 Die Funktion „Modell publizieren“	110
6.1.5 Szenario 4 : Aktualisierung eines Modells	112
6.2 Systemsicht	115
6.2.1 Codierung der Publikationsstruktur und -gestaltung	115
6.2.2 Apache Cocoon als Transformationswerkzeug	116
6.2.3 Szenario 5: Funktion “System konfigurieren“ - Erweiterung der Modellinhalte	118
6.2.4 Checkliste zur Funktion „System installieren“	121
7 Schlussbetrachtungen	123
7.1 Ergebnisüberblick	123
7.2 Weiterführende Aspekte	124
7.2.1 Werkzeugunabhängiger Einsatz mittels XMI	124
7.2.2 Verknüpfung mit Workflow-Systemen	124
7.2.3 Geschäftsprozessmanagement ist Informationsmanagement	125
Quellenverzeichnis	127
Glossar.....	131
A. Übersicht der vorgeschlagenen Modellierungskonventionen	135
B. Codierung des XML-Schemas.....	139
C. Grafische Darstellung des XML-Schemas.....	141

Abbildungsverzeichnis

Abbildung 1: Prozessdokumentation als Schnittstelle zwischen Modell und Realität.....	14
Abbildung 2: Generierung von Prozessdokumentationen in verschiedenen Formaten.....	16
Abbildung 3: Konventionen und Publikationssystem als Mittel der Prozessdokumentation...	17
Abbildung 4: Geschäftsprozess versus Workflow (Gadatsch 2002, S. 31).....	22
Abbildung 5: Zyklus des kontinuierlichen Verbesserungsprozesses (vgl. Goesmann 1998, S.1)	24
Abbildung 6: Einsatzzwecke von Prozessmodellen (Becker 2000, S. 53).....	27
Abbildung 7: Business Reengineering versus GPO (Gadatsch 2002 S. 17)	29
Abbildung 8: Übersicht der Nutzergruppen von Prozessmodellen.....	32
Abbildung 9: UML-Diagramme und ihre prinzipielle Verwendung.....	36
Abbildung 10: UML-Anwendungsfall-Diagramm.....	38
Abbildung 11: UML-Klassendiagramm mit Relationen	39
Abbildung 12: UML-Klassendiagramm mit Assoziationen.....	40
Abbildung 13: UML-Aktivitätsdiagramm.....	41
Abbildung 14: UML-Aktivitätsdiagramm mit Synchronisationsbalken (Ausschnitt).....	42
Abbildung 15: UML-Aktivitätsdiagramm mit Schwimmbahnen (Ausschnitt).....	42
Abbildung 16: UML-Sequenzdiagramm.....	43
Abbildung 17: Geschäftsprozesse des IPP-Anwendersupport	50
Abbildung 18: Prozessablauf „Arbeitsplatz einrichten“.....	51
Abbildung 19: Prozessablauf „Anwender schulen“	52
Abbildung 20: Prozessablauf „Anwender unterstützen“	53
Abbildung 21: Prozessablauf "Dokumentation bereitstellen"	54
Abbildung 22: Prozessablauf "System pflegen".....	56
Abbildung 23: Objektstruktur des Request-Trackers bei IPP	58
Abbildung 25: Gliederung von Anwendungsfällen und Diagrammen in <i>Rational Rose</i>	63
Abbildung 26: Aufgabe mit obligatorischen und optionalen Teilaufgaben	65
Abbildung 27: Separate Definition von Kenntnisklassen zur Erstellung von Tätigkeitsprofilen	68
Abbildung 28: Gliederung von Klassen und Klassendiagrammen in <i>Rational Rose</i>	70
Abbildung 29: Klassen und Methoden bei der Darstellung von Systemoberflächen.....	71
Abbildung 30: Kongruenz von Aufgaben und Aktivitäten	72
Abbildung 34: Nachricht mit Datenübermittlung über Argumente.....	75
Abbildung 36: Szenario mit Nachrichten verschiedener Detailebenen.....	76
Abbildung 37: Szenario für die Bearbeitung eines Datensatzes unter Ausblendung der Systemoberfläche.....	77
Abbildung 38: Benennung von Anwenderfunktionen in Sequenzdiagrammen	78
Abbildung 39: Funktionen der Anwendersicht des Publikationssystems	84
Abbildung 40: Navigation und Gliederungsbereiche der Systemoberfläche	87
Abbildung 41: Zugriffe der Systemoberfläche auf Anleitungen, Dokumente und Printversionen.....	88
Abbildung 42: Angepasste Benennungen im Publikationssystem	89
Abbildung 43: Unterstützung einer intuitiven Orientierung und Nutzung der Systemoberfläche.....	90
Abbildung 44: Inhaltsgliederung einer Printversion	92
Abbildung 45: Ablaufstruktur der Funktion „Modell lesen“	93
Abbildung 46: Szenario 1, Schritte 1 und 2	94
Abbildung 47: Szenario 1, Schritte 3 und 4	95
Abbildung 48: Szenario 1, Schritte 5-7	96
Abbildung 49: Szenario 1, Schritt 8.....	96
Abbildung 50: Szenario 2, Schritte 1 und 2	97
Abbildung 51: Szenario 2, Schritte 3 und 4	98
Abbildung 52: Szenario 2, Schritt 5	98

Abbildung 53: Szenario 2, Schritte 6 bis 8	99
Abbildung 54: Szenario 2, Schritte 9 bis 11	100
Abbildung 55: Szenario 2, Schritte 12 und 13	100
Abbildung 56: Szenario 3, Schritte 1 und 2	101
Abbildung 57: Szenario 3, Schritte 3 bis 5	102
Abbildung 58: Szenario 3, Schritt 6.....	102
Abbildung 59: Funktionen der Datensicht und Systemsicht.....	103
Abbildung 60: Beispiel für die Struktur von XML-Daten	105
Abbildung 61: Aufbau eines XML-Elements	105
Abbildung 62: Modelldaten gemäß dem Standard XMI 1.1 (Auszug).....	107
Abbildung 63: Modelldaten gemäß dem proprietären XML-Schema (Auszug).....	109
Abbildung 64: Ablaufstruktur der Funktion „Modell publizieren“	110
Abbildung 65: Objektstruktur der Datengenerierung	111
Abbildung 66 : Szenario 4, Schritt 1.....	113
Abbildung 67: Szenario 4, Schritte 2 und 3	113
Abbildung 68: Szenario 4, Schritte 4 bis 7	114
Abbildung 69: Die vier Betrachtungs-Ebenen eines Webangebots bei Cocoon (Apache 2002)	116
Abbildung 70: Objektstruktur der Daten-Transformation	117
Abbildung 71: Szenario der Verarbeitung einer URI-Anfrage in Cocoon.....	117
Abbildung 72: Szenario 5, Schritte 1 und 2.....	118
Abbildung 73: Szenario 5, Schritte 3 und 4.....	119
Abbildung 74: Szenario 5, Schritt 5.....	120
Abbildung 75: Szenario 5, Schritt 6.....	120
Abbildung 76: Strukturbaum des proprietären XML-Schemas	141

Abkürzungsverzeichnis

ARIS	Architektur Integrierter Informationssysteme
CAD	Computer-Aided Design
CSS	Cascading Style Sheets
DESY	Deutsches Elektronen Synchrotron
DIN	Deutsches Institut für Normung
DTD	Document Type Definition
EDMS	Engineering Data Management System
eEPK	erweiterte Ereignisgesteuerte Prozesskette
eERM	erweitertes Entity-Relationship-Modell
EN	European Norm
GPO	Geschäftsprozessoptimierung
HTML	Hypertext Markup Language
IPP	Informationsmanagement, Prozesse, Projekte
ISO	International Standardization Organization
IT	Informationstechnologie
MOF	Meta-Objekt Facility
OLE	Object Linking and Embedding
OMG	Object Management Group
SVG	Scalable Vector Graphics
PDF	Portable Document Format
UML	Unified Modeling Language
URI	Universal Resource Identifier
VKD	Vorgangskettendiagramm
WKD	Wertschöpfungskettendiagramm
XMI	XML Metadata Interchange
XML	eXtensible Markup Language
XPath	XML Path Language
XSL	XML Stylesheet Language
XSL-FO	XML Stylesheet Language - Formatting Objects
XSLT	XML Stylesheet Language - Transformation

1 Einleitung

1.1 Kontext

Kommerzielle und nicht-kommerzielle Organisationen stehen vor der stetig wachsenden Herausforderung, die Erbringung ihrer Leistung effizient und kostensparend zu gestalten. Gleichzeitig soll die Qualität der angebotenen Produkte und Dienstleistungen gesteigert oder zumindest gehalten werden.

Vorrangige Aufgabe des Managements einer Organisation dabei ist es, den Bestand der Organisation zu sichern und die laufende Leistungserstellung zu optimieren. Lange Zeit bediente sich das Management dabei einer streng vertikal gegliederten, hierarchischen Organisationsstruktur, die sich an den verschiedenen betrieblichen Funktionen orientierte (z.B. Forschung & Entwicklung, Fertigung, Verkauf, Marketing, Buchhaltung etc.). Diese Sichtweise ist im Zuge der Industrialisierung unter dem Stichwort „Taylorismus“ bekannt geworden. Das Schwergewicht der Anstrengungen lag darauf, die einzelnen Organisationseinheiten (Abteilungen) optimal zu gestalten (vgl. dazu Becker 2000, S. 2ff.).

Der Fokus auf die Optimierung funktionaler Spezialisierung vernachlässigt jedoch die funktionsübergreifenden Zusammenhänge bei der Leistungserbringung. So entstehen Reibungsverluste durch unzureichend definierte Schnittstellen zwischen den an einer Leistung beteiligten funktionalen Einheiten. Seit den späten achtziger Jahren wird daher unter dem Schlagwort „business process reengineering“ ein Konzept der betrieblichen Umgestaltung von Organisationen propagiert, das sich konsequent an der Gestaltung von Geschäftsprozessen orientiert.

Im Zuge dieses Wandels von einer vertikalen, d.h. vorrangig funktional gegliederten zu einer horizontalen, d.h. prozessorientierten Organisationsstruktur werden die verschiedenen Funktionen einer Leistungserbringung von der Entwicklung bis zum Absatz vereint, indem effiziente Geschäftsprozesse definiert sowie Prozessteams mit gemischten Kompetenzen und Zuständigkeiten zusammengestellt werden (vgl. Osterloh 1999, S.22ff.). Diese Form der Umstrukturierung erleichtert eine kundenorientierte Arbeitsweise, die für eine erfolgreiche Leistungserstellung unentbehrlich geworden ist. So definieren Geschäftsprozesse häufig Vorgänge, die durch Kundenkontakte ausgelöst werden; mehr Mitarbeiter werden unmittelbar in kundenbezogene Tätigkeiten eingebunden.

Für die Umsetzung von Geschäftsprozessen im betrieblichen Alltag spielen Informationssysteme eine große Rolle. Sie automatisieren und optimieren den Informationsfluss und den Prozessablauf der einzelnen Vorgänge.

Die Aufgaben der Definition, Einführung und kontinuierlichen Optimierung von Geschäftsprozessen werden unter dem Begriff Geschäftsprozessmanagement zusammengefasst. Dabei dienen Geschäftsprozessmodelle (z.B. in grafischer Form) dazu, betriebliche Vorgänge und Strukturen für Management, IT-Verantwortliche, Mitarbeiter und Kunden transparent und nachvollziehbar zu machen. Dadurch wird es möglich, Spielräume und Schwachstellen in finanzieller, personeller und organisatorischer Hinsicht innerhalb der Prozesse zu erkennen, zu messen und im Rahmen der Optimierung zu berücksichtigen. Die auf konzeptioneller Ebene definierten Prozesse werden mit Hilfe des Workflow-Managements in konkrete Bearbeitungs- und Kommunikationsschritte übersetzt und informationstechnisch realisiert.

Ein Prozessmodell soll demnach das Vorgehen in der betrieblichen Realität optimieren, während diese die Vorgaben des Modells realisiert. Für die effektive Rückkoppelung zwischen Modell und Realität muss eine geeignete Schnittstelle in Form einer Prozessdokumentation vorhanden sein. Die folgende Abbildung verdeutlicht diesen Zusammenhang.

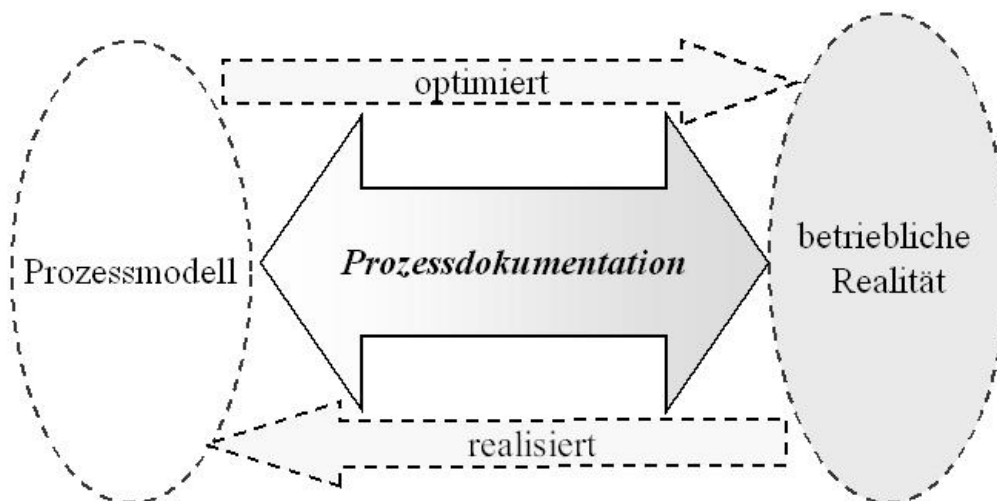


Abbildung 1: Prozessdokumentation als Schnittstelle zwischen Modell und Realität

1.2 Ziele und Ergebnisse der Arbeit

Diese Arbeit richtet sich vor allem an Verantwortliche und Ausführende in Organisationen, die sich aktiv mit der Gestaltung ihrer Geschäftsprozesse beschäftigen. Sie beleuchtet die spezifischen Aufgaben, die in diesem Zusammenhang bei der Aufbereitung und Kommunikation von Geschäftsprozessmodellen durch das betriebliche Informationsmanagement zu bewältigen sind.

Vorrangiges Ziel dabei ist es, ein Verfahren der Prozessdokumentation zu entwickeln, das Modelle bedarfsgerecht und effizient publiziert. Im Einzelnen soll das Verfahren folgende Anforderungen erfüllen:

- Die Prozessdokumentation soll das Management einer Organisation dabei unterstützen, Arbeitsabläufe und Ressourceneinsatz zu überblicken und das Optimierungspotenzial einzuschätzen.
- Die Prozessdokumentation soll für diejenigen Mitarbeiter, die im Arbeitsalltag die im Modell dargestellten Prozess-Aktivitäten ausführen, konkrete Hilfen und Arbeitsmaterialien bereitstellen. Damit soll eine direkte Schnittstelle zwischen Modell und betrieblicher Realität geschaffen werden.
- Geschäftsprozessmodelle sollen auch für Zielgruppen transparent und zugänglich gemacht werden, die keine Vorkenntnisse über Methoden und Notation der Modellierung besitzen.
- Der einzelne Publikationsvorgang soll effizient gestaltet sein, d.h. Zeitaufwand und Komplexität der Handhabung müssen in einem angemessenen Verhältnis zum Umfang und zur Bedeutung des jeweiligen Modells stehen.
- Das Verfahren soll wiederverwendbar sein, d.h. es soll unabhängig von den spezifischen Festlegungen eines bestimmten Modells nutzbar sein.

Das in dieser Arbeit vorgestellte Verfahren der Prozessdokumentation beinhaltet zwei wesentliche Mittel:

Zum einen werden Modellierungskonventionen erarbeitet, die eine einheitliche und konsistente Erstellung von Prozessmodellen auf Basis der Unified Modeling Language mit dem Modellierungswerkzeug *Rational Rose* ermöglichen.

Die Konventionen treffen u.a. Aussagen zu folgenden Bereichen:

- Basisstruktur des Modells;
- Anwendung von UML-Diagrammen und -Sprachelementen;
- Logische Verknüpfung der Modellsichten;
- Aufbau von Handlungsanweisungen;
- Modellierung von Systemoberflächen;
- Einbindung von Arbeitsmitteln.

Zum anderen wird ein Publikationssystem vorgestellt, mit dem Prozessdokumentationen parallel sowohl in interaktiv navigierbarer als auch in linearer, ausdrückbarer Form erstellt werden können (s. Abbildung 2).

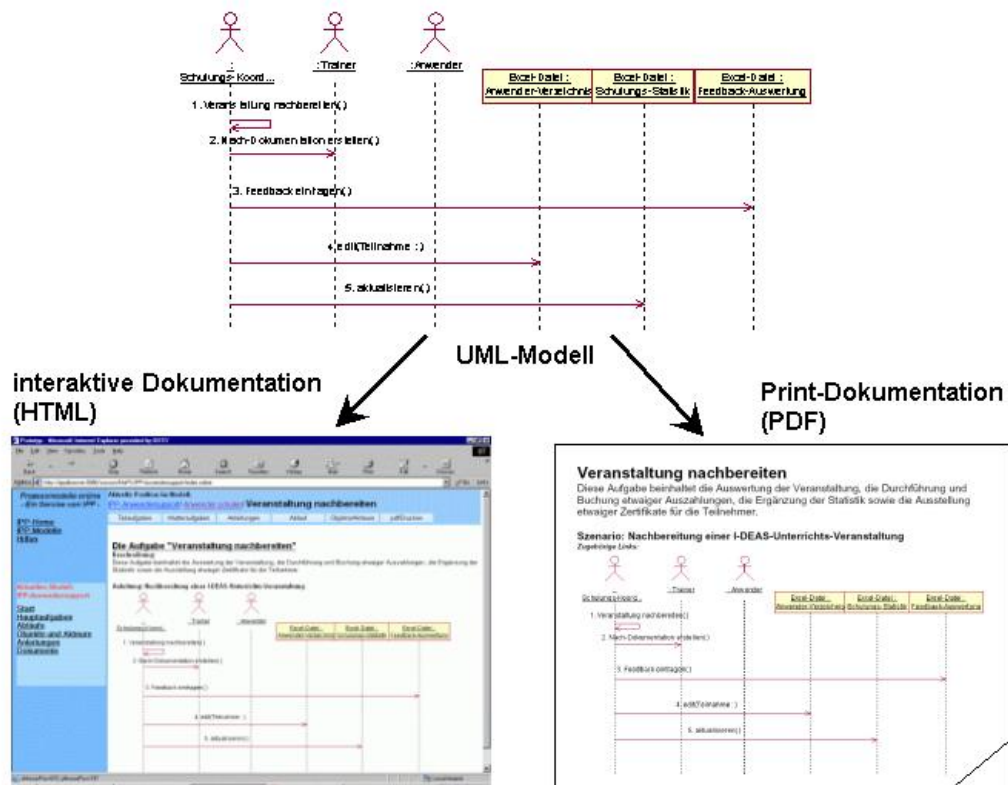


Abbildung 2: Generierung von Prozessdokumentationen in verschiedenen Formaten

Das System generiert die Dokumentationen auf Basis der Modellierungskonventionen und mit Hilfe der XML-Technologie.

Es ermöglicht u.a. die intuitive Navigation zwischen den Modellsichten und die Ansicht der Tätigkeitsprofile von Akteuren. Darüber hinaus bietet es den Zugriff auf Arbeitsmittel, die im betrieblichen Alltag verwendet werden.

Beide Mittel – Konventionen und Publikationssystem – sind aufeinander abgestimmt und bilden zusammen die Brücke zwischen Prozessmodell und betrieblicher Realität (s.Abbildung 3).

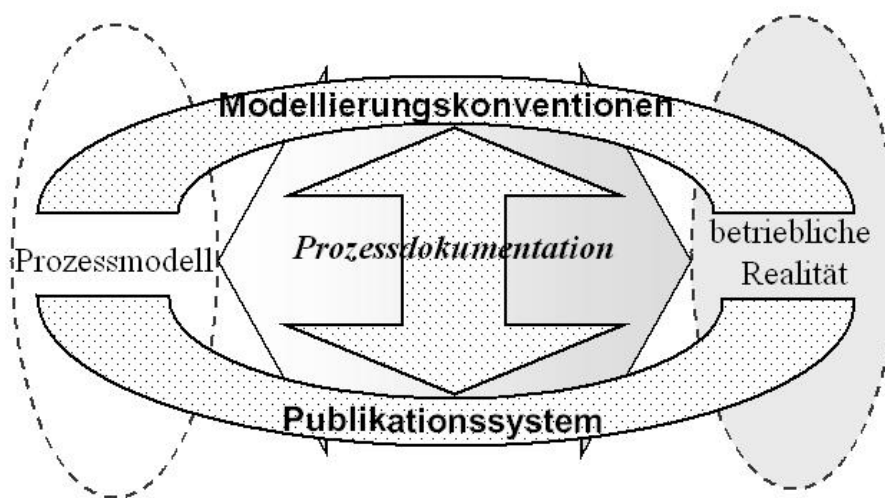


Abbildung 3: Konventionen und Publikationssystem als Mittel der Prozessdokumentation

1.3 Vorgehensweise und Gliederung

Für das grundsätzliche Verständnis der Arbeit ist es zunächst erforderlich, die Bedeutung einiger Begriffe zu bestimmen und weitere Grundlagen der Geschäftsprozessmodellierung zu erörtern. Dabei werden auch die Eigenschaften und Anforderungen der Nutzer von Geschäftsprozessmodellen näher bestimmt. Nach der Definition von Modellsichten wird die prinzipielle Anwendung der Unified Modeling Language bei der Modellierung von Geschäftsprozessen vorgestellt (Kapitel 2).

Der konkrete Bezugsrahmen der weiteren Arbeit wird durch ein Fallbeispiel bestimmt. Der Autor hat im Zuge eines Projektes beim Deutschen Elektronen-Synchrotron (DESY) das Geschäftsprozessmodell eines betriebsinternen Anwendersupports überarbeitet und dokumentiert. Auf dieses konkrete Modell wird im weiteren Verlauf der Ausführungen wiederholt Bezug genommen, um Anforderungen und Verwendung des Dokumentationsverfahrens anhand von Beispielen zu veranschaulichen. Dazu werden Kontext und Geschäftsprozesse des Fallbeispiels erläutert (Kapitel 3).

Die zielgruppengerechte Kommunikation eines Modells macht es erforderlich, dass schon bei seiner Erstellung auf eine bedarfsgerechte und anschauliche Aufbereitung geachtet wird. Dazu gehört vor allem die einheitliche und konsistente Anwendung der Modellierungssprache und des Modellierungswerkzeugs. Ausgehend von den verschiedenen Modellsichten und anhand wiederkehrender Fragestellungen bei der Modellierung werden Vorschläge für Modellierungskonventionen für die Anwendung der UML mit *Rational Rose* entwickelt (Kapitel 4).

Im weiteren Verlauf der Arbeit wird das Publikationssystem vorgestellt, mit dem Prozessdokumentationen auf Basis der Modellierungskonventionen generiert werden können.

Die Anwendersicht des Publikationssystems beinhaltet die bedarfsgerechte und gebrauchstaugliche Darstellung der Geschäftsprozessmodelle. Es wird die parallele Dokumentation eines Modells im Intranet sowohl in einer interaktiv navigierbaren Version als auch in linearer, ausdrückbarer Form erörtert. Hinsichtlich der Gestaltung

und Funktionalität des Publikationssystems wird die Berücksichtigung anerkannter Kriterien der Gebrauchstauglichkeit erläutert. Der praktische Nutzen des Systems wird anhand von Anwenderszenarien demonstriert (Kapitel 5).

Datensicht und Systemsicht des Publikationssystems verbinden die spezifische Funktionalität des ausgewählten Modellierungswerkzeugs *Rational Rose* mit den Sprachen und Methoden der XML-Technologie. Die Verwendung eines proprietären XML-Schemas wird vorgestellt und die Funktionsweise des Systems bei der Publizierung eines Modells wird erläutert.

Es wird gezeigt, wie die Prozessdokumentationen mit Hilfe des Webservlets *Cocoon 2* von *Apache* generiert werden. Dazu wird die Funktionsweise von *Cocoon* bei der Datentransformation vorgestellt. Die Konfiguration des Systems wird anhand eines Beispiels erörtert. (Kapitel 6).

Die Schlussbetrachtungen enthalten neben einer Zusammenfassung und Bewertung einen Ausblick auf weiterführende Aspekte bei einer möglichen Optimierung und Erweiterung des Dokumentationsverfahrens.

2 Grundlagen der Geschäftsprozessmodellierung

2.1 Management von Geschäftsprozessen

2.1.1 Definition des Geschäftsprozesses

Im alltäglichen Gebrauch der Unternehmens- und Fachwelt ist der Begriff „Geschäftsprozess“ nicht mit großen Unsicherheiten verbunden, d.h. in der Regel besteht eine stillschweigende Übereinkunft darüber, was er meint und in welchem Zusammenhang er betrachtet werden muss. Geht es aber konkret um die Erstellung eines Geschäftsprozessmodells, werden die genauen Abgrenzungen zu anderen Begriffen und Darstellungs-Gegenständen der Modellierung relevant.

Die ISO-Begriffsdefinition im Rahmen der Normen für das Qualitätsmanagement definiert einen Prozess als einen

„Satz von in Wechselbeziehungen stehenden Mitteln und Tätigkeiten, die Eingaben in Ergebnisse umgestalten“

(NORM 1997).

Darauf aufbauend werden folgende Definitionen der weiteren Arbeit zu Grunde gelegt.

Geschäftsprozess – Ein Geschäftsprozess ist ein Prozess, der im Wesentlichen innerhalb einer Organisation stattfindet und der Verwirklichung eines Organisationsziels dient.

Kernprozess – Ein Kernprozess ist ein Geschäftsprozess, dessen Ergebnis unmittelbar eine Verwirklichung eines Organisationsziels darstellt (Die Begriffe **Schlüsselprozess**, **Hauptprozess** und **Kernprozess** werden in diesem Kontext als synonym angesehen).

Unterstützungsprozess - Ein Unterstützungsprozess ist ein Geschäftsprozess, dessen Ergebnis der Verwirklichung des Organisationsziels nur mittelbar dient. (Die Begriffe **Unterstützungsprozess** und **Supportprozess** werden in diesem Kontext synonym verwendet).

Teilprozess – Ein Teilprozess ist ein Prozess, der innerhalb eines anderen Prozesses vollständig ausgeführt wird.

Vorgang/Geschäftsvorfall – Beide Begriffe werden synonym verwendet. Sie bezeichnen die Instanz eines Prozesses, also das reale Ereignis eines Prozessablaufs.

Aufgabe/Funktion – Beide Begriffe werden synonym verwendet. Sie stehen für die Gliederung bzw. Beschreibung einer Organisation hinsichtlich ihrer Ziele und Teilziele, d.h. eine Aufgabe oder Funktion entspricht einem (Teil-)Ziel.

Aktivität – Eine Aktivität ist ein Schritt innerhalb eines Prozesses und ist damit die triviale Form eines Teilprozesses. Sie kann aber auch selbst wiederum aus mehreren Aktivitäten bestehen, die gemeinsam einen Teilprozess formen.

Aktion – Eine Aktion ist eine elementare Aktivität, die keine weitere Untergliederung besitzt.

Operation/Methode – Dies ist die für ein Objekt oder einen Akteur definierte Fähigkeit zur Ausführung einer Aktion oder Aktivität.

2.1.2 Geschäftsprozesse und Workflows

Geschäftsprozesse beschreiben den Ablauf wesentlicher Aufgaben, die für die Zielerreichung der gesamten Organisation entscheidend und prägend sind. Die konkrete Ausgestaltung der Geschäftsprozesse erfordert eine detaillierte Beschreibungsebene, die die Aufgaben in einzelne operative Tätigkeiten aufschlüsselt und den zuständigen Akteuren bzw. IT-Systemen in einer bestimmten Abarbeitungsreihenfolge zuordnet. Diese verfeinerte Definition von Prozessschritten nennt man Workflow.

Dokumentierte Workflows können einerseits in Form von Arbeitsanweisungen für die Prozessakteure und andererseits in Form von Sequenzen von Funktionsaufrufen der beteiligten Anwendungssysteme eingesetzt werden.

Die folgende Tabelle stellt Geschäftsprozess- und Workflow-Ebene einander gegenüber und verdeutlicht die jeweilige spezifische Bedeutung.

	<i>Geschäftsprozess</i>	<i>Workflow</i>
Ziel	Analyse und Gestaltung von Arbeitsabläufen im Sinne gegebener (strategischer) Ziele	Spezifikation der technischen Ausführung von Arbeitsabläufen
Gestaltungsebene	Konzeptionelle Ebene mit Verbindung zur Geschäftsstrategie	Operative Ebene mit Verbindung zu unterstützender Technologie
Detaillierungsgrad	In einem Zug von einem Mitarbeiter an einem Arbeitsplatz ausführbare Arbeitsschritte	Konkretisierung von Arbeitsschritten hinsichtlich Arbeitsverfahren sowie personeller und technologischer Ressourcen

Abbildung 4: Geschäftsprozess versus Workflow (Gadatsch 2002, S. 31)

Das Management der dargestellten zwei Ebenen ist jedoch in der Praxis nicht strikt voneinander zu trennen. Vielmehr kann das Workflow-Management als ein Teilbereich des umfassenden Geschäftsprozess-Managements betrachtet werden, mit dem die Übergänge vom Geschäftsmodell zur betrieblichen Umsetzung definiert werden.

2.1.3 Aufgaben des Geschäftsprozessmanagements

Organisationen, die ihr betriebliches Geschehen konsequent an Geschäftsprozessen ausrichten, müssen eine Reihe von Aufgaben berücksichtigen, die unter dem Stichwort Geschäftsprozessmanagement folgendermaßen zusammengefasst werden können:

- Darstellung der Prozesse
- Optimierung der Prozesse
- Simulation der Prozesse
- Automatisierung betrieblicher Abläufe
- Informationstechnische Realisierung der Prozesse

(Bucher 2001, S. 22f.)

Eingeführte Prozesse werden im Rahmen einer kontinuierlichen Geschäftsprozessoptimierung ständig an dem Anspruch einer effizienten Erreichung definierter Organisationsziele sowie an den sich wandelnden externen Anforderungen gemessen und entsprechend angepasst.

Bei diesen Aufgaben spielen Modelle der Geschäftsprozesse eine große Rolle. Sie liefern eine vereinheitlichte, idealtypische Beschreibung des betrieblichen Geschehens und dienen als Vorlage für die Durchführung der Geschäftsvorfälle. Modelle erlauben es, bei Störungen im Betriebsalltag systematisch nach Ursachen und Lösungsmöglichkeiten zu suchen. Sie dienen als Diskussionsgrundlage für das theoretische Durchspielen sowohl von Standard- als auch Ausnahmesituationen, um Spielräume für Optimierungen entdecken und nutzen zu können. Die Modelle schaffen auch die Voraussetzungen für die Entwicklung von Informationssystemen, die die Geschäftsprozesse unterstützen, einzelne Vorfälle dokumentieren oder in Form umfassender Workflowmanagement-Systeme alle Prozessschritte der Vorgänge praxisnah in einer gemeinsamen IT-Anwendung integrieren und transparent machen.

Geschäftsprozessmanagement besteht daher zu einem großen Teil aus der Erstellung, Implementierung und Optimierung von Geschäftsprozessmodellen.

Der in Abbildung 5 dargestellte Zyklus zeigt die prinzipiellen Phasen bei der kontinuierlichen Optimierung der Modelle. Sie gelten analog sowohl für die Geschäftsprozess- als auch die Workflow-Ebene.

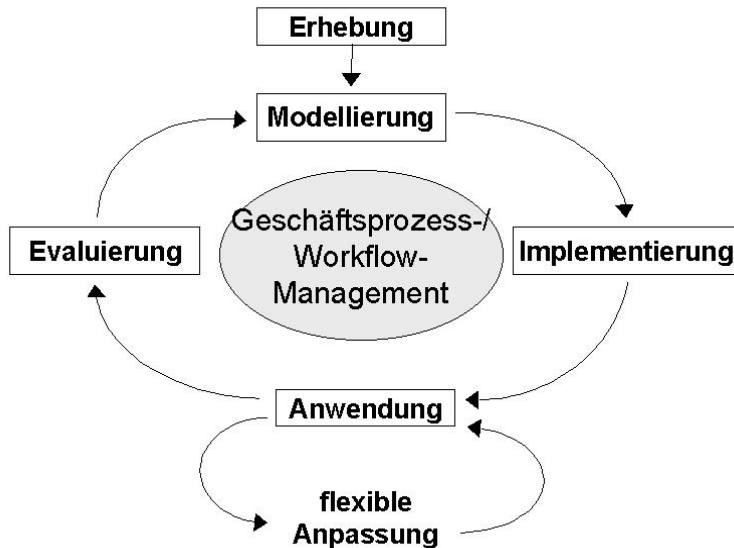


Abbildung 5: Zyklus des kontinuierlichen Verbesserungsprozesses (vgl. Goesmann 1998, S.1)

Sowohl für die Erstellung und Optimierung eines Geschäftsprozessmodells, als auch für seine gelungene Anwendung im betrieblichen Alltag sind geeignete technische, methodische und personelle Voraussetzungen in der jeweiligen Organisation zu schaffen. Je nach Einsatzbereich eines Modells bzw. je nach der gerade durchlaufenen Phase im Optimierungszyklus werden dabei unterschiedliche Nutzergruppen mit unterschiedlichen Anforderungen an Inhalt, Detailtiefe und Präsentation des Modells angesprochen (s. Abschnitt 2.2).

Diese Nutzergruppen benötigen einen gemeinsamen Bezugsrahmen sowie eine Informationsquelle für die Modellinhalte. Beides muss durch eine zielgruppengerechte Aufbereitung und Kommunikation der Prozessmodelle im Rahmen einer Prozessdokumentation sichergestellt werden.

2.1.4 Die Rolle des betrieblichen Informationsmanagements

"Gegenstandsbereich des Informationsmanagements ist die effektive und effiziente Bewirtschaftung des Produktionsfaktors Information in Organisationen."

(Herget 1997)

Informationen sind wesentliche Objekte in Geschäftsprozessen, als Auslöser, Hilfsmittel oder Produkt der Vorgänge. Geschäftsprozesse sind somit zu einem großen Teil Informationsprozesse.

Ebenso spielt die Entwicklung und der Betrieb von Informationssystemen eine zentrale Rolle bei der Implementierung der Geschäftsprozesse. Informationssysteme werden an den Geschäftsprozessen der Organisation ausgerichtet und unterstützen den Informationsfluss der Geschäftsvorfälle. Vor diesem Hintergrund ist die Erstellung und Optimierung der Geschäftsprozessmodelle auch eine Aufgabe des Informationsmanagements.

Bei der Aufbereitung und Kommunikation von Geschäftsprozessmodellen nimmt das betriebliche Informationsmanagement darüber hinaus in folgenden Bereichen eine Schlüsselrolle ein:

Visualisierung: Eine effektive Kommunikation der Modelle ist wesentlich für die praktische Umsetzung der Geschäftsprozesse. Für die Beschreibung des Modells muss daher eine Notation bzw. Sprache gewählt werden, die für alle Nutzer verständlich bzw. schnell zu erlernen ist. Daneben gehört vor allem die überschaubare und logische Präsentation der Modelle mit Hilfe geeigneter Publikationen in gedruckter und elektronischer Form zu den typischen Aufgaben des Informationsmanagements.

Systematisierung: Die Definition einer einheitlichen Begriffswelt zur Beschreibung der den Prozess bestimmenden Aufgaben, Akteure und Klassen ist eine unentbehrliche Voraussetzung, deren Erfüllung eine klassische Aufgabe des Dokumentationswesens darstellt.

Vor allem bei der Modellierung von Akteuren, Zuständigkeiten, Aufgaben und Begriffen ist daher auf konsistente Definitionen und Beschreibungen zu achten, die Widersprüche und Redundanzen vermeiden. Auch dies dient der Verständlichkeit, bietet aber darüber hinaus die Möglichkeit, die verschiedenen Geschäftsprozesse zu integrieren und aufeinander optimal abzustimmen, um einen effizienten Betrieb zu gewährleisten.

Standardisierung: Geschäftsprozessmodelle sollten nach einheitlichen Regeln erstellt werden, um ein gemeinsames Verständnis der Beteiligten innerhalb der Organisation zu erleichtern und die Anwendung (halb-)automatischer Dokumentationsverfahren zu ermöglichen. Diese Regeln sollten einerseits detailliert genug sein, dass alle wesentlichen Muster und Situationen der jeweiligen betrieblichen Wirklichkeit abgebildet werden können, andererseits nicht so kompliziert sein, dass den Autoren der Geschäftsmodelle unangemessene Hürden bei der Modellierung und späteren Anpassung entstehen.

Daneben ist auch die Auswahl und spezifische Anwendung eines Modellierungswerkzeugs ein notwendiger Schritt der Standardisierung.

Ebenso wie die Modelle sollten auch ihre Dokumentationen einheitlich gestaltet sein, um die wiederholte Verwendung als Referenz und Diskussionsgrundlage zu erleichtern.

Entwicklung von Informationssystemen: Gerade die Dokumentation von komplexen Geschäftsprozessmodellen kann durch die Bereitstellung eines entsprechenden Publikationssystems effizienter durchgeführt werden. Ein solches System muss gebrauchstaugliche Funktionen bieten, die den Anforderungen der verschiedenen Nutzergruppen gerecht werden. Es muss so gestaltet sein, dass dem Nutzer die schrittweise Annäherung an das Geschäftsprozessmodell und dessen intuitive Nutzung ermöglicht wird.

Im Rahmen der technischen Implementierung müssen ein geeignetes Datenformat und die Methoden für den Datenexport, die Datentransformation und die Publikationsgenerierung bestimmt werden. Zudem müssen die verschiedenen Systemkomponenten an die gegebene IT-Struktur der Organisation angepasst und installiert werden.

2.2 Verwendung von Geschäftsprozessmodellen

Die Verwendung von Prozessmodellen kann entsprechend Abbildung 6 in Zwecke einerseits bei der Organisationsgestaltung und andererseits bei der Anwendungssystemgestaltung aufgeschlüsselt werden.

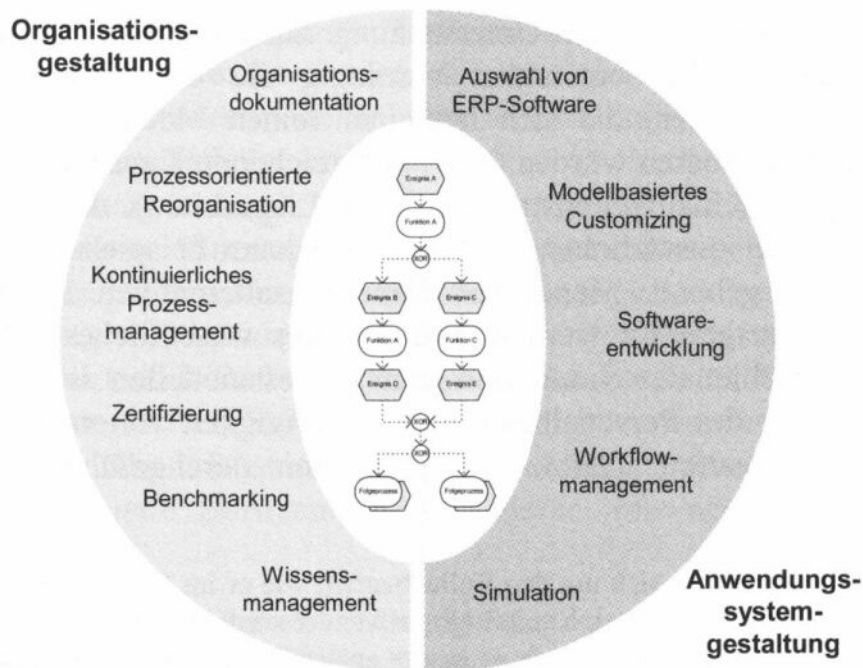


Abbildung 6: Einsatzzwecke von Prozessmodellen (Becker 2000, S. 53)

Für diese Anwendungsbereiche sind vor allem die betriebliche Führung sowie die Funktionsbereiche Qualitätsmanagement und IT-Management Nutzer von Prozessmodellen. Darüberhinaus nutzen die prozessausführenden und -verantwortlichen Mitarbeiter die Prozessmodelle im Kontext ihres jeweiligen Aufgabenbereichs.

Auch Kunden und (Geschäfts-)Partner sind mögliche Nutzer von Prozessmodellen.

Die Sichtweisen und Anforderungen der verschiedenen Verwendungsbereiche und Nutzergruppen werden im Folgenden dargelegt.

2.2.1 Strategisches Management

Im Rahmen des strategischen Managements wird die grundsätzliche Ausrichtung einer Organisation gestaltet. Dazu gehören vor allem die Definition der Geschäftsfelder und die Herausbildung einer Organisationskultur und -struktur, die der Erreichung der Organisationsziele dient.

Die dabei festgelegten Strategien sind der Ausgangspunkt für das zu erstellende oder zu optimierende Geschäftsprozessmodell. Das strategische Management hat die Aufgabe, die Verknüpfung zwischen diesen Grundlinien und dem Vorgang der Modellierung sicherzustellen. Diese Aufgabe existiert in zwei unterschiedlichen Ausprägungen:

Im Falle eines Business Reengineering wird eine Organisation komplett neu gestaltet, es kommt zu einem radikalen Wandel, bei dem eine Organisation meist zum ersten Mal konsequent an Geschäftsprozessen ausgerichtet wird.

Besteht bereits ein Prozessmodell, findet eine kontinuierliche Geschäftsprozessoptimierung statt, bei der in kleinen überschaubaren Schritten Anpassungen am Modell bzw. seiner betrieblichen Anwendung vorgenommen werden. Diese Anpassungen orientieren sich an veränderten Umweltbedingungen und innerbetrieblichen Entwicklungen.

Geschäftsprozessmodelle bilden dabei eine strukturierte Projektionsfläche und dokumentierte Diskussionsgrundlage für die Weiterentwicklung einer Organisation. Erst mit einem umfassenden Modell haben die beteiligten Akteure des strategischen Managements eine Kommunikationsbasis für die gesamte Bandbreite ihrer Entscheidungen.

Die folgende Abbildung stellt die Zielrichtung der Konzepte des Business Reengineering und der Geschäftsprozessoptimierung (GPO) einander gegenüber:

	Business Reengineering	Geschäftsprozessoptimierung
Wirkung auf die existierende Organisation	Tiefgreifende Veränderung Ersatz der alten Organisation Völlige Neukonzeption	Verbesserung der bestehenden Organisation
Veränderung der Organisation	Quantensprünge des Wandels, d.h. radikale Veränderung	Organisationsentwicklung auch in kleinen Schritten Moderate Veränderung
Methode der Prozessbeschreibung	Prozessverstehen, d.h. Verzicht auf Details	Prozessanalyse durch formale detaillierte Beschreibung

Abbildung 7: Business Reengineering versus GPO (Gadatsch 2002 S. 17)

2.2.2 Qualitätsmanagement

Die Qualität der Leistungserstellung einer Organisation ist neben dem Preis entscheidend für ihren Fortbestand und Erfolg. Sie misst sich vor allem am Grad der Zufriedenheit der Kunden mit den erbrachten Leistungen. Die Qualitätssicherung ist daher eine zentrale Aufgabe des Managements. Qualitätsmanagement betrachtet aber nicht allein die Qualität der angebotenen Leistungen oder Produkte, sondern umfasst alle Ebenen einer Organisation, d.h. alle Strukturen und Tätigkeiten werden an Qualitätsmaßstäben ausgerichtet. So setzen qualitativ hochwertige Leistungen auf Dauer auch effektiv und effizient gestaltete Geschäftsprozesse voraus.

Eine große Rolle spielen auch Kenngrößen, die versuchen die Leistungsfähigkeit einer Organisation zu messen. Ein Großteil dieser Kenngrößen existiert in Form von finanzwirtschaftlichen und ertragswirtschaftlichen Kennzahlen, die den wirtschaftlichen Zustand des Gesamtunternehmens erfassen und im Rahmen des betrieblichen Rechnungswesens erhoben werden (vgl. dazu Wöhe 2000, S. 1083ff.). Kenngrößen können aber nicht nur für die wirtschaftliche Dimension definiert werden, sondern auch für Bereiche der Leistungserbringung wie die technische Leistungsfähigkeit, Vertriebseffizienz und Planerfüllung (vgl. dazu Specht 2000, S. 569ff.).

Dazu gehören beispielsweise die Kosten eines Produkts oder einer Dienstleistung, die Antwort- und Reaktionszeiten bei externen Kontakten oder die umgeschlagene Menge der Leistungen.

Die zuverlässige und standardisierte Erfassung von Kenngrößen ist eine entscheidende Voraussetzung für ihre Aussagekraft. Grundlage für die Erfassung ist die Erstellung und Dokumentation eines umfassenden Geschäftsmodells, das die Ziele der Organisation definiert sowie die Struktur hinsichtlich der Aufbauorganisation und Geschäftsprozesse festlegt. Innerhalb des Modells können die Geschäftsobjekte und Prozessschritte ausgewählt werden, die für die Bestimmung einer Kenngröße entscheidend sein sollen.

Kenngrößen sind dazu geeignet aufzuzeigen, dass es Schwächen in der Leistungserbringung der Organisation gibt. Wo diese Schwächen liegen, wird durch den Kontext der Kenngröße in den jeweiligen Geschäftsprozessen deutlich. Eine qualitativ hochwertige Dokumentation des Geschäftsmodells ermöglicht die Kommunikation des Modells innerhalb der Organisation und erleichtert die Suche nach den Ursachen für etwaige Schwächen. Somit wird die überwiegend quantitative Betrachtungsweise über Kenngrößen durch eine qualitative Sicht auf die Leistungserbringung der Organisation erweitert.

Der Zusammenhang zwischen Qualitätsmanagement und Geschäftsprozessen manifestiert sich auch in den Anforderungen der Zertifizierung nach DIN ISO 9000ff.. Organisationen, die diese Zertifizierung anstreben, müssen ihr Qualitätssicherungssystem (QS) dokumentieren. Dazu gehört auch die Aufnahme entsprechender Verfahrens- und Arbeitsanweisungen. Diese können aus den Prozessmodellen abgeleitet werden, wozu jedoch eine entsprechende Detailebene vorhanden sein bzw. erstellt werden muss (vgl. dazu Becker 2000, S. 283ff.).

2.2.3 IT-Management

Die Beschreibung einer Organisation über Geschäftsprozesse ist auch eine Folge der rasanten Entwicklung bei der Nutzung von Informationstechnologie. Immer mehr Organisationen sind in allen Bereichen ihrer alltäglichen Arbeit von einer funktionsfähigen Informationstechnik abhängig, die die einzelnen Schritte der Geschäftsprozesse unterstützt bzw. dokumentiert. Dennoch sind Geschäftsprozesse nicht etwas wirklich Neues in Organisationen. Vielmehr werden bestehende Abläufe neu beschrieben und transparent gemacht, wo früher die Optimierung der Aufbaustruktur im Vordergrund stand. Die zunehmende Entwicklung hoch-komplexer Informationssysteme, die auf die Besonderheiten einer ganz bestimmten Organisation

abgestimmt sind, hat wesentlich dazu beigetragen, die Anforderungen und Arbeitsschritte in neuer Weise systematisch zu erfassen und zu beschreiben.

Besonders bei der Entwicklung und Anpassung spezialisierter Workflow-Management-Systeme ist eine umfassende Beschreibung der Geschäftsprozesse erforderlich, die für die einzelnen Prozessaktivitäten eine detaillierte Definition der Workflows mit einschließt. Ein entsprechend umfangreiches und detailliertes Modell dient dann als Grundlage für die Definition und Gewichtung der Anforderungen an das einzuführende System. Ein hauseigenes System kann gezielt auf diese Anforderungen hin entwickelt werden. Bestehende Systeme von Softwareherstellern können an diesen Anforderungen im Rahmen von Benchmark-Test gemessen und damit ihre Eignung bewertet werden.

2.2.4 Mitarbeiter

Nicht nur auf der Management-Ebene, auch auf der Ebene der ausführenden Mitarbeiter haben Geschäftsprozesse eine Relevanz. Sie sind die Akteure, die durch ihr betriebliches Wirken ein Prozessmodell mit Leben füllen.

Das im Geschäftsprozessmodell dokumentierte Wissen über Ziele, Aufbau und Arbeitsweise einer Organisation hilft dem einzelnen Mitarbeiter seine Rolle zu definieren und sich mit seiner Aufgabe und der Gesamtorganisation zu identifizieren. Beinhaltet ein Modell auch Beschreibungen auf der Workflow-Ebene, können diese als Handlungsanweisungen für die alltägliche Praxis verwendet werden. Eine entsprechend ausgerichtete Modelldokumentation kann somit als Referenz-Quelle für die verschiedensten betrieblichen Standardsituationen verwendet werden.

Was für eingearbeitetes und routiniertes Personal eine hilfreiche Ergänzung darstellt, ist für neu einzuarbeitendes Personal eine unentbehrliche Grundlage für die Orientierung über das eigene Tätigkeitsprofil. Gerade beim Umgang mit bisher unbekanntem Anwendungssystemen können Workflow-Szenarien hilfreiche Hinweise für den Einstieg bereitstellen.

Darüberhinaus sollte auch der mögliche Beitrag der Mitarbeiter zur Optimierung der Geschäftsprozesse nicht unterschätzt werden. Anschaulich dokumentierte Modelle schaffen eine gemeinsame Diskussionsgrundlage zwischen Management und ausführenden Mitarbeitern. Mängel in ihrer Arbeitssituation und Verbesserungsvorschläge können somit durch die Mitarbeiter präziser und zielgerichteter formuliert werden.

2.2.5 Kunden und Geschäftspartner

Geschäftsprozesse beschreiben Vorgänge zur Erbringung einer Leistung für Kunden. Die Kundenorientierung ist also ein prägendes Merkmal des Geschäftsprozessmanagements. Dabei muss nicht nur der Prozess selbst an den Bedürfnissen des Kunden ausgerichtet werden, auch die Information über den Prozess, dessen Ergebnis der Kunde erhält, ist ein Teil dieser Kundenorientierung.

Eine Organisation erhöht ihre Vertrauenswürdigkeit, wenn sie ihre internen Abläufe und Strukturen zumindest soweit offen legt, dass deren Sinn erkennbar wird. Dies zielt nicht nur auf die bestehenden und potenziellen Kunden, sondern auch auf Geschäftspartner. Diese können anhand der dokumentierten Modelle den Bedarf an Leistungen sowie Möglichkeiten der Zusammenarbeit besser einschätzen und ihre Angebote und Leistungen darauf abstimmen.

2.2.6 Übersicht der Nutzergruppen

Die folgende tabellarische Übersicht stellt die verschiedenen Nutzergruppen mit ihren Anwendungsbereichen der Prozessmodelle zusammenfassend dar. Daraus können die Anforderungen an die Verwendung der Modellierungssprache (s. Kapitel 4) sowie an das Publikationssystem (s. Kapitel 5 und 6) abgeleitet werden.

<i>Nutzergruppe</i>	<i>Ziele und Verwendung der Modelle</i>
Strategisches Management	<ul style="list-style-type: none"> ○ Ausrichtung auf strategische Geschäftsziele ○ Optimierung von Struktur und Leistung ○ Strategisches Controlling
IT-Management	<ul style="list-style-type: none"> ○ Entwicklung und Anpassung von IT-Systemen ○ Benchmark-Tests
Qualitätsmanagement	<ul style="list-style-type: none"> ○ Definition und Messung von Kenngrößen ○ Schwachstellen-Analyse ○ Integration von Qualitätssicherungs-Maßnahmen ○ Zertifizierung
Mitarbeiter	<ul style="list-style-type: none"> ○ Definition und Identifikation mit der eigenen Rolle ○ Förderung vernetzten Denkens und Handelns
- Prozessverantwortliche	<ul style="list-style-type: none"> ○ Optimierung der Arbeitsorganisation ○ Überwachung der Vorgänge
- Prozessausführende	<ul style="list-style-type: none"> ○ Konkrete Arbeitsanweisung ○ Bedienung von IT-Systemen ○ Checklisten
- neue Mitarbeiter	<ul style="list-style-type: none"> ○ Einarbeitung und Schulung
Kunden, Geschäftspartner	<ul style="list-style-type: none"> ○ Transparenz der Organisation ○ Einschätzung der Qualität ○ Optimierung externer Leistungen

Abbildung 8: Übersicht der Nutzergruppen von Prozessmodellen

2.3 Modellierung von Geschäftsprozessen

2.3.1 Inhalt und Darstellung von Geschäftsprozessmodellen

Geschäftsprozesse werden durch Modelle beschrieben, die den Ablauf wiederkehrender Vorgänge vereinheitlichen und Alternativen für Situationen aufzeigen, die nicht dem Regelfall entsprechen. Anhand der Modelle wird auch das Aufgabenprofil einer Organisation und ihrer Akteure sowie der Einsatz von Arbeitsmitteln beschrieben und transparent gemacht. Dabei können Modelle nie vollständig alle Aspekte der Prozesse und ihres Kontextes bis in alle Einzelheiten abbilden; es findet immer ein gewisses Maß von Vereinfachung statt. Diese Vereinfachung bewegt sich auf dem schmalen Grat zwischen notwendiger Genauigkeit und gewünschter Anschaulichkeit der Modelle und muss beide Anforderungen ausbalancieren und miteinander in Einklang bringen.

Für die Darstellung von Prozessmodellen sind sowohl seitens der Betriebswirtschaft als auch seitens der Informatik visuelle Notationen und Sprachen entwickelt und bereitgestellt worden, die einerseits einen schnellen Überblick über einen Ablauf bzw. eine Struktur erlauben sollen, andererseits für die Spezifikation von Informationssystemen detaillierte Vorgaben beinhalten können. Gegenüber einer rein verbalen Modellbeschreibung haben sie den Vorteil, komplexe Zusammenhänge in einer unmittelbaren Form darstellen zu können. Als Nachteil ist ihnen gemeinsam, dass sie bei ihrer Einführung erst neu erlernt werden müssen. Neben der Notation bestimmter Diagrammtypen existieren auch umfassende Modellierungsmethoden, die mehrere Modellsichten integrieren. Zwei umfassende und weit entwickelte Methoden sind die Architektur Integrierter Informationssysteme (ARIS) und die Unified Modeling Language (UML).

Der Schwerpunkt und die Stärke von ARIS liegt in der strategischen und semantischen Konzeption eines Geschäftsprozessmodells und ist dadurch eher betriebswirtschaftlich geprägt (vgl. dazu Scheer 1998, Gadatsch 2002). Die UML wurde für die Planung und Spezifikation von Softwaresystemen entwickelt, beinhaltet aber auch starke Ausdrucksmöglichkeiten für die betriebswirtschaftliche Geschäftsprozessmodellierung (s Abschnitt 2.4).

Die konkrete Anwendung einer Modellierungssprache muss an die Anforderungen der jeweiligen Organisation angepasst werden. Dabei besteht ein Bedarf nach

einheitlichem Vorgehen bei der Modellierung und nach Einhaltung bestimmter Qualitätsstandards. Beides dient dazu, den Nutzen der Modelle bei möglichst geringem Aufwand zu maximieren. Für diesen Zweck sind - angelehnt an die Grundsätze ordnungsmäßiger Buchführung - folgende Grundsätze ordnungsmäßiger Modellierung entwickelt worden (vgl. Becker 2000, S. 65ff.; Scheer 1998, S. 119f.):

- Grundsatz der Richtigkeit
- Grundsatz der Relevanz
- Grundsatz der Wirtschaftlichkeit
- Grundsatz der Klarheit
- Grundsatz der Vergleichbarkeit
- Grundsatz des systematischen Aufbaus

Aufbauend auf diesen Grundsätzen können spezifische Modellierungskonventionen für einen bestimmten Anwendungskontext definiert werden (s. Kapitel 4).

2.3.2 Bestimmung von Modellsichten

Die Geschäftsprozesse stellen nur eine Sicht auf das Modell einer Organisation dar. Ein umfassendes Geschäftsmodell beschreibt eine Organisation hinsichtlich aller wesentlichen Sichtweisen. Es umfasst die Zieldefinition, sowohl die Ablauf- als auch die Aufbauorganisation sowie die Rolle beteiligter Geschäftsobjekte wie Produkte, Dokumente und IT-Systeme.

Geschäftsprozesse dürfen nicht isoliert betrachtet werden, sondern müssen im größeren Zusammenhang der Aufbauorganisation und beteiligter Geschäftsobjekte definiert werden. Erst wenn die verschiedenen Sichten auf eine Organisation bei der Modellierung berücksichtigt sind, kann ein Geschäftsprozess optimal gestaltet werden. Entscheidend ist dabei die Koordinierung der Sichten untereinander.

Je nach Autor und Modellierungsmethode werden diese Sichten verschieden benannt und teilweise auch unterschiedlich definiert und voneinander abgegrenzt (vgl. dazu Gadatsch 2002, S. 44 ff.).

Eine gängige Weise diese Sichten zu definieren wird im Folgenden erläutert. Sie wird in der Fachliteratur für die Modellierung von Informationssystemen für Geschäftsprozesse vorgeschlagen (vgl. dazu Schönsleben 2001, S.147). Da solche Informationssysteme die Geschäftsprozesse im Idealfall genau widerspiegeln, werden

die beschriebenen Sichten hier analog dazu verwandt, die Modellierung des Geschäftsprozessmodells selbst zu beschreiben.

Die vier Sichten auf eine Organisation sind demnach die Prozesssicht, die Funktionssicht, die Objektsicht und die Organisationssicht.

Die **Prozesssicht** zeigt für ein bestimmtes Ziel (Funktion) den Ablauf von Tätigkeiten mit den möglichen Verzweigungen und Bearbeitungs-Reihenfolgen. Dabei kann auch die Verwendung von Arbeitsmitteln und die Rolle wesentlicher Akteure deutlich gemacht werden. Sie ist somit die Sicht, die am dichtesten beim Arbeitsgeschehen der Mitarbeiter einer Organisation liegt.

Die **Funktionssicht** definiert Funktionen einer Organisation, indem sie Tätigkeiten, die innerhalb eines Prozesses vorkommen bzw. dem gleichen Ziel dienen, logisch zusammenfasst. Dabei werden die verschiedenen Funktionen miteinander in Beziehung gesetzt, meist in hierarchischer Form. Die obersten Funktionen können, müssen dabei aber nicht den Geschäftsprozessen der Organisation entsprechen. (Funktion und Aufgabe werden hier synonym verwendet).

Die **Objektsicht** definiert die in einer Organisation vorhandenen Daten und realen Dinge (Arbeitsmittel, Produkte etc.) als Objekte und zeigt deren Beziehungen untereinander. Den Objekten zugeordnete Attribute kennzeichnen sie im Detail. Zudem kann sie Zuordnungen von Prozessen, Funktionen und Aufgaben zu Objekten aufzeigen.

Die **Organisationssicht** fasst Prozesse, Funktionen und Tätigkeiten in einer Weise zusammen, dass daraus zusammenhängende organisatorische Einheiten entstehen. Damit vollzieht sich in der Organisationssicht der Übergang von der Ablauf- zur Aufbaustruktur einer Organisation.

2.4 Die Unified Modeling Language(UML)

2.4.1 Grundlagen der UML

Die UML ist eine objekt-orientierte, grafische Modellierungssprache, die für die Planung und Spezifikation von Softwaresystemen entwickelt wurde. Sie dient dazu, zwischen Auftraggebern, Anwendern, Software-Entwicklern und Programmierern eine gemeinsame Verständigung über Anforderungen und Funktionsweise eines Systems zu ermöglichen.

Sie entstand in der zweiten Hälfte der neunziger Jahre aus der Synthese verschiedener unabhängiger Methoden (vgl. Booch 1999, S. xviii ff.) und ist seit 1997 ein von der Object Management Group (OMG) genehmigter Standard, für den im März 2003 die Version 1.5. herausgegeben wurde (vgl. dazu OMG 2003).

Ein Vorteil der UML liegt in ihrer großen Anwendungsbreite. Durch die Konversion zahlreicher Elemente verschiedener Notationen kann sie eine Vielzahl von Modellierungs-Anforderungen sowohl von betriebswirtschaftlicher als auch informationstechnischer Seite umsetzen und miteinander verknüpfen.

Die Objektorientierung der UML beinhaltet die Kapselung und Vererbung von Verhalten und Zuständen der Modellobjekte. Daher ist sie besonders geeignet, Geschäftsprozesse in einer Weise zu beschreiben, dass sie leicht mit Hilfe von Informationssystemen realisiert werden können.

Die UML unterscheidet neun Diagrammtypen, deren prinzipielle Verwendung in der folgenden Tabelle wiedergegeben wird.

Diagrammtyp	Inhalt
Anwendungsfall-Diagramme	zeigen die Funktionen eines Systems aus der Perspektive des Anwenders
Aktivitätsdiagramme	zeigen den Ablauf von Aktivitäten eines Systems und seiner Verwendung
Zustandsdiagramme	stellen die Reihenfolge der Zustandsänderungen von Objekten dar
Sequenzdiagramme	zeigen die zeitliche Abfolge der Nachrichten, die zwischen Objekten und Akteuren gesendet werden
Kollaborations-Diagramme	zeigen Objekte und Akteure mit den Nachrichten die sie miteinander austauschen
Klassendiagramme	zeigen die Klassen mit Attributen und Methoden sowie die zwischen ihnen bestehenden Beziehungen und stellen somit den statischen Aufbau eines Systems dar
Objektdiagramme	zeigen die auf Klassendiagrammen definierten Beziehungen für konkrete Instanzen von Klassen (Objekte)
Komponenten-diagramme	zeigen die elementare Systemstruktur auf Datei- und Codeebene
Verteilungs-diagramme	zeigen die Systemarchitektur auf Netzwerk- und Hardwareebene

Abbildung 9: UML-Diagramme und ihre prinzipielle Verwendung

Neben diesen neun Diagrammen dienen Pakete, Teilsysteme und Modellsichten dazu, die Inhalte eines Modells auf konzeptioneller Ebene zu gliedern

Darüberhinaus ermöglicht die UML die selbständige Erweiterung der Notation durch den Modellierer mit Hilfe von frei definierbaren Stereotypen, Eigenschaften und Einschränkungen. Dadurch kann sie an die spezifischen Anforderungen eines bestimmten Modells angepasst werden.

Die UML selbst ist eine Sprache, die als solche keine Vorgaben darüber beinhaltet, mit welcher Methodologie bestimmte Arten von Modellen erstellt und dargestellt werden (vgl. OMG 2002). Die genaue Verwendung der Symbole und Elemente muss sich im Kontext der jeweiligen Organisation herausbilden, so dass verschiedene, parallel existierende Modellierungs-Stile entstehen können. Die im folgenden erläuterte Verwendung der UML wird bei IPP seit einigen Jahren praktiziert und wird in ähnlichen Varianten auch in der Fachliteratur für die Geschäftsprozessmodellierung vorgeschlagen (vgl. dazu Grässle 2002; Eriksson 2002, S. 57f.). Dabei wird das in Abschnitt 2.3.2 erläuterte Sichtenkonzept zugrunde gelegt.

2.4.2 Darstellung der Funktionssicht

Hierzu werden Anwendungsfälle verwendet. Dabei entspricht ein Anwendungsfall einem ganzen Geschäftsprozess oder einer Funktion, die innerhalb eines Geschäftsprozesses ausgeführt wird. Anwendungsfall-Diagramme zeigen den Zusammenhang zwischen den Geschäftsprozessen und den dabei verwendeten Funktionen. Diese Zusammenhänge werden vor allem durch zwei Relationen ausgedrückt: Die Relation „verwendet“ sagt aus, dass ein Anwendungsfall obligatorischer Teil bei der Ausführung eines anderen Anwendungsfalles ist. Demgegenüber definiert die Relation „erweitert“ den Zusammenhang, dass ein Anwendungsfall optionaler Teil eines anderen Anwendungsfalles ist. Die folgende Abbildung zeigt ein Anwendungsfall-Diagramm, das für einen Prozess „Arbeitsplatz einrichten“ sieben Teilfunktionen definiert.

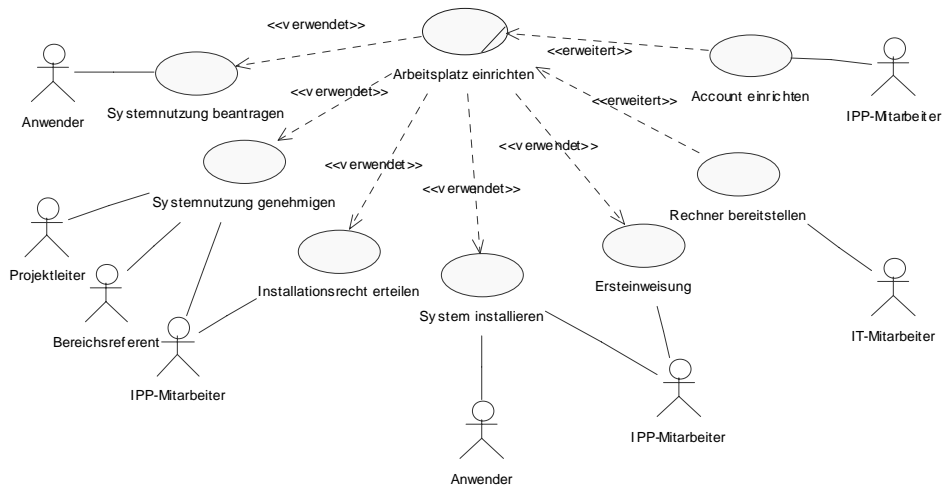


Abbildung 10: UML-Anwendungsfall-Diagramm

Wie im Beispiel zu sehen, können auf Anwendungsfall-Diagrammen auch die Rollen bzw. Akteure gezeigt werden, die an einer Funktion bzw. einem Geschäftsprozess maßgeblich beteiligt sind.

2.4.3 Darstellung der Organisationssicht (Prozess-Beteiligte)

Die beteiligten Personen, Abteilungen oder Instanzen in einem Geschäftsprozess werden durch das Symbol „Akteur“ repräsentiert. Die Akteure werden dabei hinsichtlich ihrer Rolle in der Organisation oder auch gegenüber der jeweiligen Aufgabe definiert. Sie sind diejenigen, sie in einem Geschäftsprozess aktiv handeln, entweder als Auslöser, Zulieferer oder Erbringer einer Leistung. Für Akteure können Verhalten und Eigenschaften definiert werden. Die Beziehungen von Akteuren untereinander und zu Objekten werden in Klassendiagrammen dargestellt.

2.4.4 Darstellung der Objektsicht (Hilfsmittel und Produkte)

Arbeits-Werkzeuge, Datenspeicher, Dokumente und Arbeitsergebnisse sind Dinge des realen Kontextes eines Geschäftsprozesses. Sie werden in der UML als Objekte definiert. Mehrere Objekte gleicher Art bilden dabei eine Klasse. Verhalten und Eigenschaften von Objekten werden durch die Methoden und Attribute der Klassen, zu denen sie gehören, definiert. Die Beziehungen von Objekten bzw. Klassen werden in Klassendiagrammen dargestellt (s.u.).

Neben den Klassendiagrammen bietet die UML mit den Komponenten- und Verteilungsdiagrammen zusätzliche Möglichkeiten, die Struktur und das Verhalten

von Informationssystemen im Detail zu modellieren. Somit kann sowohl für die Modellierung von Geschäftsprozessen als auch für die Entwicklung beteiligter Software-Systeme der gleiche Standard verwendet werden.

2.4.5 Darstellung von Beziehungen und Strukturen

Die Beziehungen zwischen den Objekten und Akteuren eines Geschäftsprozesses werden über verallgemeinerte Beziehungen zwischen den jeweiligen Klassen definiert. Diese Beziehungen werden auf einem Klassendiagramm dargestellt, das damit einen Überblick über die Struktur einer Organisation oder zumindest eines Ausschnitts davon bietet. Die folgende Abbildung zeigt das Beispiel einer Objekt-Struktur.

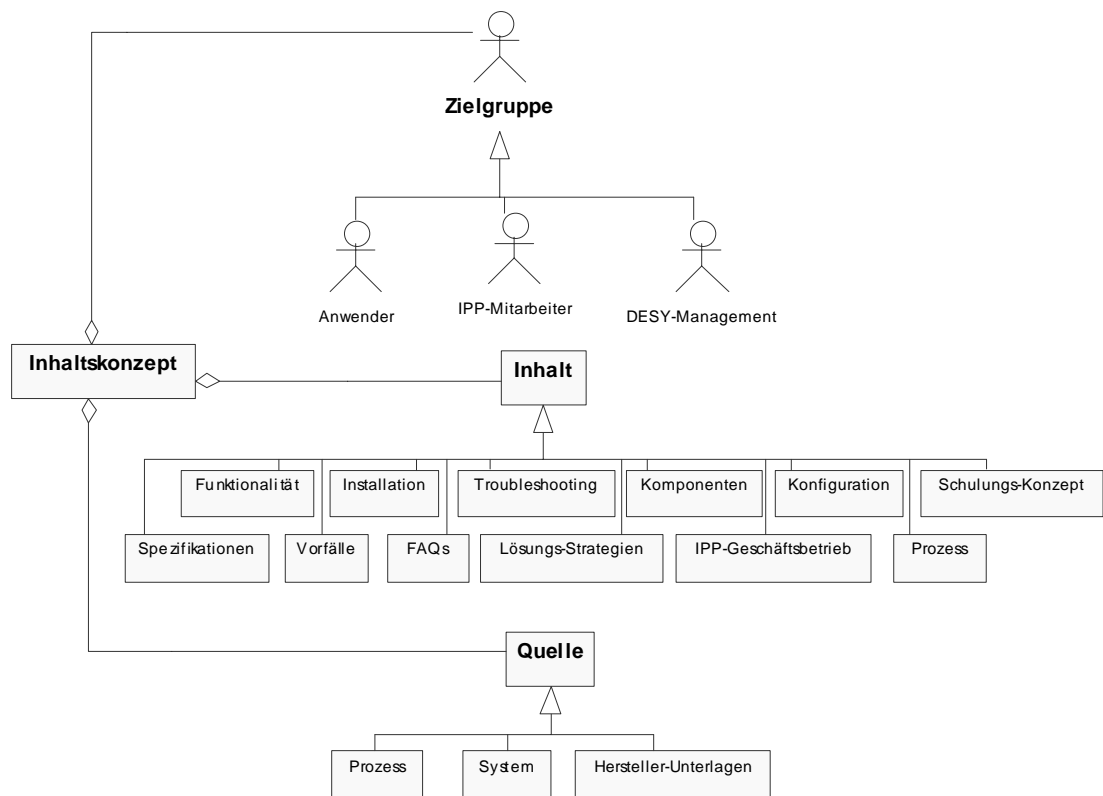


Abbildung 11: UML-Klassendiagramm mit Relationen

In der Abbildung sind zwei wesentliche Typen von Relationen erkennbar: Aggregation („Teil-Ganzes“-Beziehung) und Vererbung („Spezialisierung-Generalisierung“-Beziehung). Wörtlich lässt sich das Diagramm folgendermaßen übersetzen: Ein Inhaltskonzept besteht aus Quellen, Inhalten und Zielgruppen. Quellen können Prozesse, Systeme oder Hersteller-Unterlagen sein; Zielgruppen können Anwender, IPP-Mitarbeiter oder DESY-Management sein; Inhalte können

u.a. Vorfälle, Komponenten oder Schulungskonzepte sein. Dieses Klassendiagramm legt nicht fest, welche Inhaltskonzepte konkret existieren, sondern es zeigt auf abstrakter Ebene, wie Inhaltskonzepte prinzipiell aufgebaut sind und welche Möglichkeiten der Bildung eines Objekts vom Typ „Inhaltskonzept“ bestehen. Ein konkretes Inhaltskonzept wären z.B. Lösungsstrategien, die für Anwender auf der Grundlage von Hersteller-Unterlagen dokumentiert werden.

Eine weitere Beziehungstyp zwischen Klassen und Akteuren ist die Assoziation. Sie kann eine Richtung haben oder ungerichtet sein, und zeigt zunächst auf, dass zwei Klassen oder Akteure etwas „miteinander zu tun“ haben. Welche konkrete Bedeutung eine Assoziation tatsächlich hat, kann durch einen Namen und ggf. weitere Angaben relativ frei definiert werden. Ein Beispiel wird mit der nächsten Abbildung gegeben.

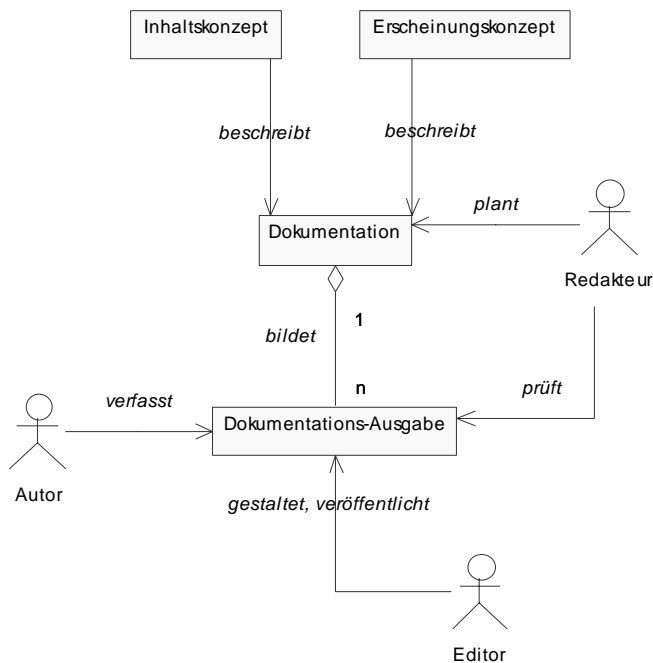


Abbildung 12: UML-Klassendiagramm mit Assoziationen

Pfeile bei Relationen zwischen Klassen machen die vorrangige Wirkungsrichtung der Beziehung deutlich, beschreiben jedoch keinen Ablauf.

2.4.6 Darstellung der Prozesssicht auf Geschäftsprozessebene

Ein Geschäftsprozess wird zwar auch durch statische Elemente beschrieben, der Kern seiner Definition liegt jedoch im Prozess-Ablauf. Die UML bietet hierfür eigene Elemente an: Aktivitäten und Übergänge. Aktivitäten repräsentieren die Ausführung

einer oder mehrerer Funktionen; Übergänge verbinden Aktivitäten in einer Richtung und bestimmen so deren Reihenfolge.

Wie im folgenden Beispiel des Prozessablaufs „Arbeitsplatz einrichten“ deutlich wird, hat ein Prozessablauf genau einen Startpunkt, kann aber mehrere Endpunkte haben.

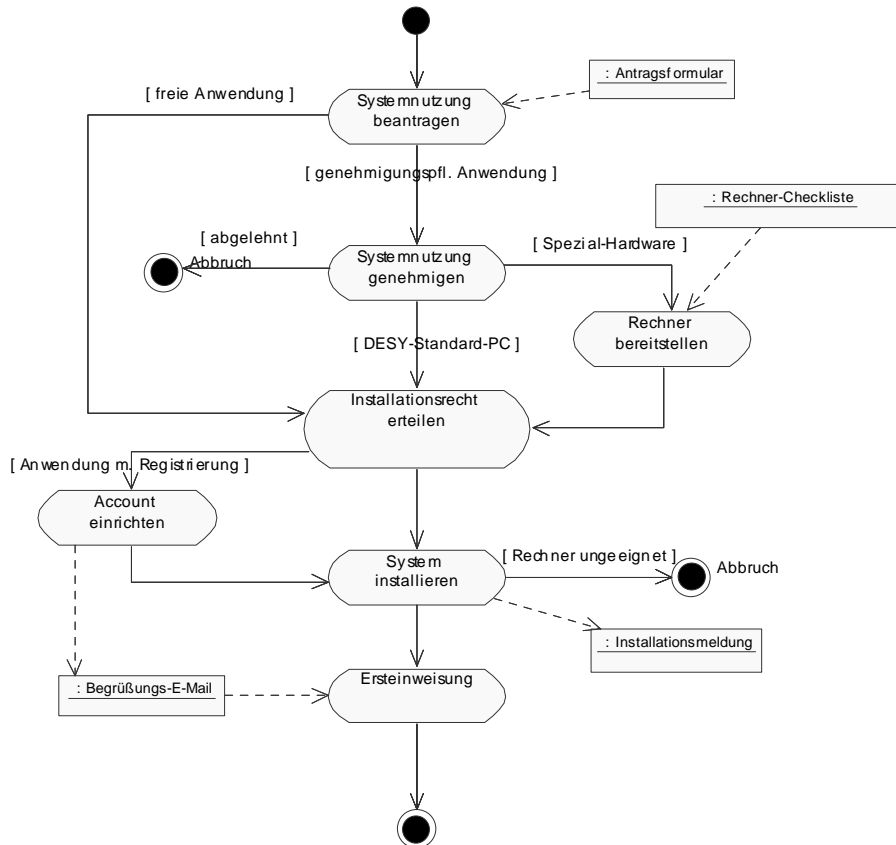


Abbildung 13: UML-Aktivitätsdiagramm

Ein Übergang kann mit einer Bedingung verknüpft sein, die erfüllt sein muss, damit der Übergang aktiv wird. So ist im Beispiel zu erkennen, dass der Übergang zur Aktivität „Rechner bereitstellen“ nur aktiv wird, wenn die Bedingung „Spezial-Hardware“ erfüllt ist. An dieser Stelle ist auch zu sehen, dass eine Aktivität durchaus mehrere Übergangsausgänge und -eingänge haben kann, wodurch Verzweigungen möglich werden. Dabei werden die verschiedenen Übergänge nur alternativ aktiv. Demgegenüber werden die Übergänge von oder zu Aktivitäten, die parallel ausgeführt werden sollen, durch Synchronisations-Balken zusammengeführt.

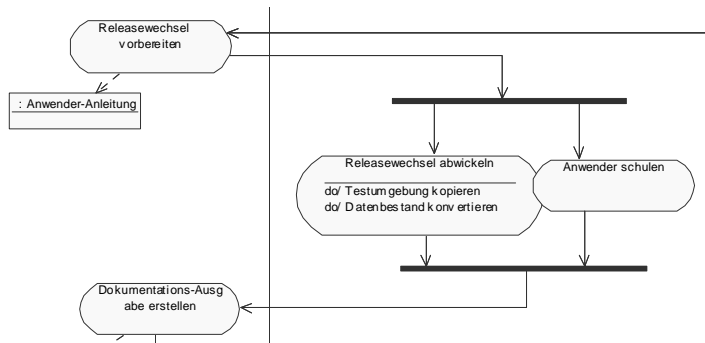


Abbildung 14: UML-Aktivitätsdiagramm mit Synchronisationsbalken (Ausschnitt)

Durch eine Gliederung des Diagramms in Spalten – sogenannte Schwimmbahnen – können die Aktivitäten beteiligten Akteure bzw. Klassen zugeordnet werden. Dies dient vor allem dazu, die Zuständigkeiten von Akteuren aber auch ihre Zusammenarbeit deutlich zu machen.

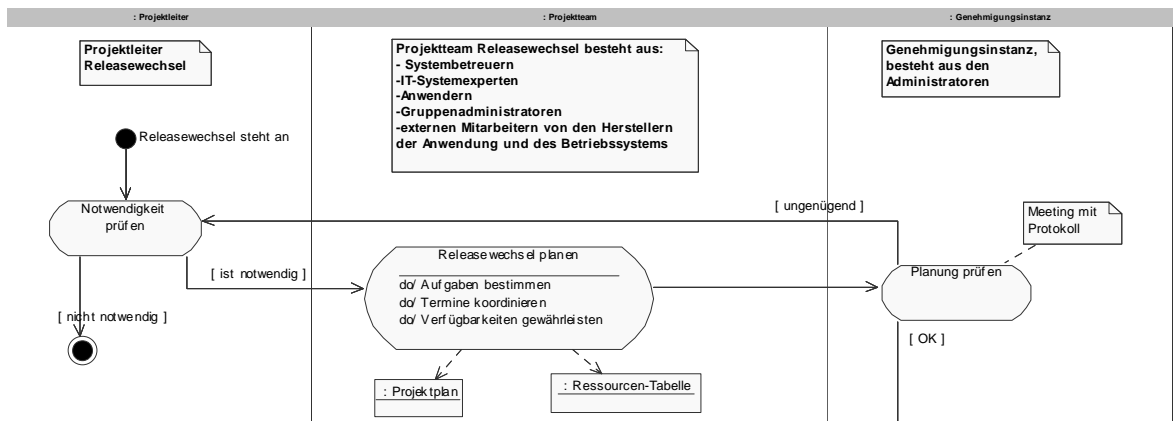


Abbildung 15: UML-Aktivitätsdiagramm mit Schwimmbahnen (Ausschnitt)

2.4.7 Darstellung der Prozesssicht auf Workflowebene

Das vorherige Beispiel zeigt die mögliche Komplexität eines Prozesses. So gibt es mehrere Wege durch den Prozess, je nach Erfüllung entsprechender Übergangsbedingungen. Die Aktivitäten sind teilweise zu grob definiert, um den beteiligten Akteuren die einzelnen Arbeitsschritte verdeutlichen oder die Funktionalität verwendeter Informationssysteme abbilden zu können. Zu diesem Zweck werden Workflow-Szenarien erstellt.

Ein Szenario konkretisiert den Ablauf eines Prozesses, indem es einen bestimmten Weg durch die Verzweigungen des Prozesses aufzeigt oder/und weitere Detailschritte innerhalb der Aktivitäten verdeutlicht. Szenarien werden mit Hilfe von Sequenzdiagrammen erstellt. Abbildung 16 zeigt ein Sequenzdiagramm mit einem Szenario für den Ablauf des Prozesses „Arbeitsplatz einrichten“.

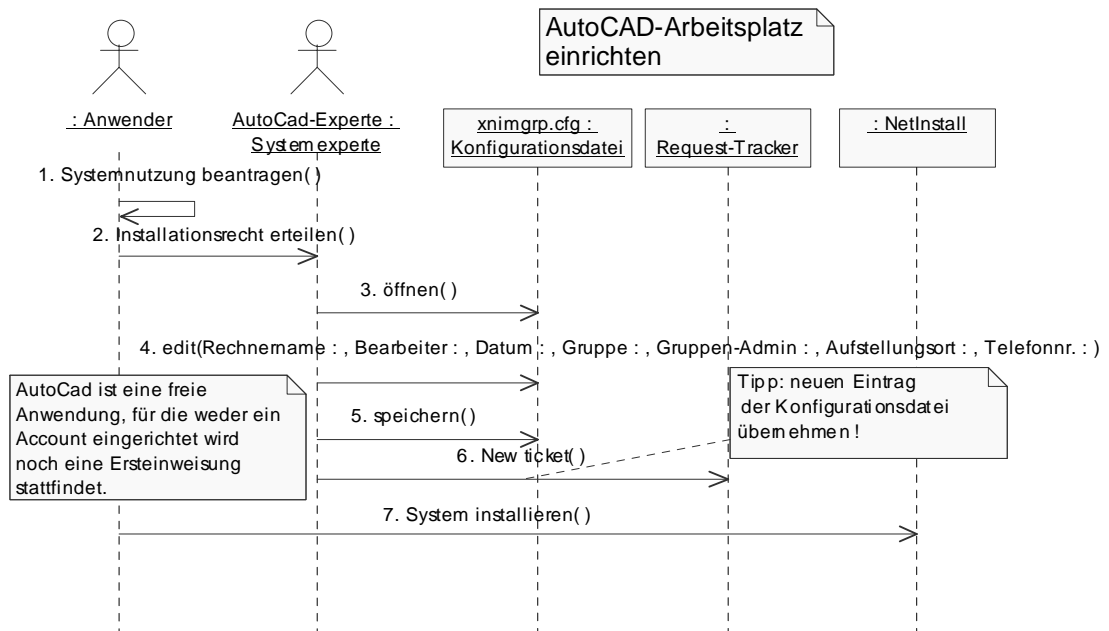


Abbildung 16: UML-Sequenzdiagramm

Ein Sequenzdiagramm zeigt dabei, wie beteiligte Akteure und Objekte miteinander kommunizieren. In horizontaler Richtung werden die Symbole der Akteure und Objekte aufgereiht; sie erhalten eine vertikal nach unten verlaufende Linie, die dem zeitlichen Verlauf entspricht. Die Kommunikation zwischen den Objekten und Akteuren wird durch Pfeile definiert, die den Nachrichten entsprechen, die Objekte und Akteure senden bzw. empfangen. Somit ist ein Sequenzdiagramm eine Möglichkeit, detaillierte Handlungsanweisungen in Prozessen zu modellieren und zu veranschaulichen.

3 Fallbeispiel: Prozessdokumentation für den DESY-Anwendersupport

3.1 Beschreibung des Projektumfangs

3.1.1 Anlass und Hintergrund des Projekts

Hintergrund dieser Arbeit ist ein Projekt der Gruppe *Informationsmanagement, Prozesse, Projekte (IPP)* des Deutschen Elektronen-Synchrotrons (DESY) in Hamburg. Sie verwendet UML-Modelle zur Beschreibung von Geschäftsprozessen ihres eigenen Aufgabenbereichs und derjenigen anderer Gruppen bei DESY, die die Unterstützung von IPP bei der Optimierung ihrer Arbeitsabläufe in Anspruch nehmen.

Was ist DESY ?

Das Deutsche Elektronen Synchrotron DESY ist eine Forschungseinrichtung der Helmholtz-Gemeinschaft Deutscher Forschungszentren (HGF) mit Standorten in Hamburg und Zeuthen. Gegründet 1959, betreibt sie Forschung im Bereich der Elementarteilchen-Physik und der wissenschaftlichen Nutzung von Synchrotron-Strahlung. DESY betreibt mehrere Teilchen-Beschleuniger und ist dadurch das führende deutsche Kompetenz-Zentrum für den Bau von Beschleuniger-Anlagen. DESY ist eine selbständige Stiftung bürgerlichen Recht mit ca. 1200 Mitarbeitern und wird ausschließlich aus öffentlichen Mitteln finanziert (vgl. DESY 2000).

Was ist IPP ?

Die DESY-Gruppe Informationsmanagement, Prozesse, Projekte (IPP) besteht seit 1998 und umfasst ca. 15 Mitarbeiter. Sie ist für die Schaffung eines integrierten Informationsmanagements verantwortlich, um dadurch die Transparenz und Abwicklung von DESY-Großprojekten zu unterstützen. Arbeitsschwerpunkte sind

- die Mitwirkung in Projekten,
- die Einführung von Informationssystemen,
- der Anwendersupport mit dem Betrieb von CAD- und Informationssystemen sowie
- die methodische Unterstützung von Projekten (vgl. DESY 2003).

Nach ihrer Gründung Ende 1998 begann IPP damit, eine prozessorientierte Arbeitsorganisation aufzubauen. Mit ihr sollte die Kommunikation verbessert, Zuständigkeiten geklärt und eine Prozesskostenrechnung ermöglicht werden. Die Prozesse wurden durch Prozessmodelle definiert, die zunächst mit der Notation Bonapart erstellt wurden und seit 2001 mit der Modellierungssprache UML (Unified Modeling Language) (s. Abschnitt 2.4) dargestellt werden.

Im Januar 2001 hatte IPP ein Dokument fertiggestellt, das die Leistungen und Prozesse des damals noch auf CAD-Systeme beschränkten Anwendersupports

beschreibt (vgl. dazu DESY 2001). Dazu wurden zu jedem Prozess Ziel, Auslöser, Qualitätsmerkmale, Abläufe und Dokumentations-Hinweise definiert. Die Abläufe wurden mit Hilfe von Aktivitäts- und Sequenzdiagrammen der UML modelliert.

Die Modelle dienen als Orientierung über die Anforderungen und Möglichkeiten der CAD-Anwender und des CAD-Supports. Mitarbeiter nutzten sie als Referenz über notwendige und optionale Arbeitsschritte bei der Abwicklung einzelner Vorgänge sowie die dabei erforderlichen Hilfsmittel. Eingearbeitete Mitarbeiter lösen sich in ihrem Arbeitsalltag jedoch von dem Anspruch, die Vorgänge zu 100 % gemäß den Modellen zu gestalten. Vielmehr spielt ein gewisser Pragmatismus in der Praxis eine große Rolle. Dies ist jedoch kein wirklicher Bruch mit den Modellen, sondern von vornherein so berücksichtigt und gewollt. Folgendes wird dazu in der Präambel des entsprechenden Dokuments ausgeführt:

Die in diesem Dokument vorgestellten Arbeitsabläufe bilden den organisatorischen Rahmen des CAD-Supports. Sie stellen keine Einschränkung der Flexibilität dar, im Zweifelsfall ist immer der gesunde Menschenverstand maßgebend.

(DESY 2001, S. 2)

Im Jahr 2002 wurde mit einer grundlegenden Überarbeitung der Prozesse begonnen, die durch eine SWOT-Analyse eingeleitet wurde. Die Überarbeitung wurde notwendig, um die Aufnahme weiterer Softwaresysteme zu ermöglichen und die Integration neuer Mitarbeiter zu erleichtern. Zudem sollten die künftige Verwendung eines Help-Desk-Systems, die Bestimmung von Prozesskennzahlen und die Umstellung der Anwenderschulung auf Inhouse-Angebote berücksichtigt werden.

Die Überarbeitung der Prozesse machte den Bedarf nach einer automatisch aktualisierbaren Prozessdokumentation deutlich, die die Prozesse nicht nur darstellt, sondern auch Zugriff auf Arbeitsmaterialien (Formulare, Checklisten etc.) bietet und die Erstellung von Tätigkeitsprofilen der Akteure unterstützt.

3.1.2 Projektziele und -produkte

Ziel des Projektes war es, die Prozesse des IPP-Anwendersupports für alle beteiligten Mitarbeiter durch geeignete Prozessdokumentationen transparent zu machen, und somit eine gemeinsame Arbeitsgrundlage sicherzustellen. Desweiteren sollte die Dokumentation einzelner Vorgänge im Anwendersupport ausgeweitet und vereinheitlicht werden. Änderungen an den Prozessmodellen und an den sie beschreibenden Publikationen sollten von den verantwortlichen Mitarbeitern selbständig durchgeführt werden können.

Konkret spiegeln sich diese Ziele in folgenden Produkten wieder, die im Laufe des Projektes erstellt wurden:

- Modelle zur Beschreibung der fünf Prozesse des IPP-Anwendersupports
- Spezifische Konventionen für die Modellierung
- Print- und HTML-Publikationen der Modelle
- Verfahren zur automatischen Generierung der Publikationen nach Änderungen
- Technische Dokumentationen

3.1.3 Angestrebter Nutzen

Mitarbeiter, die neu in den Anwendersupport eingearbeitet werden, erhalten mit den dokumentierten Prozessmodellen einen Überblick über ihre jeweilige Aufgabe. Während der laufenden Bewältigung der Aufgaben können sie die Publikationen als Referenz bei auftretenden Fragen nutzen. Der einheitliche Umgang mit Geschäftsvorfällen erleichtert die Zusammenarbeit und die Verständigung untereinander, so dass sich Mitarbeiter ohne größere Hindernisse gegenseitig vertreten können.

Die Veröffentlichung im Internet verschafft den Anwendern einen Einblick in die Bearbeitung ihrer Anfragen und ist somit ein Beitrag zur Transparenz des Anwendersupports.

Die Leitung von IPP und die Prozessverantwortlichen nutzen die Modelle, um die Prozesse zu optimieren. Zudem stellen die Modelle zusammen mit dem Monitoring entsprechender Prozesskennzahlen eine geeignete Diskussions-Grundlage für die Erarbeitung von Service-Level-Agreements dar.

3.1.4 Vorgehensweise bei der Modellierung

Die bisherigen Prozesse wurden mit den beteiligten Mitarbeitern in einer SWOT-Analyse bewertet. Aufgaben und Arbeitsmittel wurden mit Hilfe von Metaplan-Methoden gesammelt und sachlich und chronologisch geordnet. Anschließend wurden die Prozesse mit Hilfe der UML neu modelliert und dokumentiert. Überwiegend bauen die Modelle auf den bereits bestehenden Entwürfen auf und verfeinern sie. Ein Teil der Prozesse wurde vollkommen neu modelliert, um neuen Anforderungen gerecht zu werden. Die neuen Modell-Entwürfe wurden anschließend in Zusammenarbeit mit der Leitung von IPP und einzelnen jeweils zuständigen Mitarbeitern diskutiert, ggf. überarbeitet und ergänzt, bevor sie zur Publikation freigegeben wurden.

Schließlich wurden relevante Arbeitsmittel zur Durchführung der Prozesse zusammengeführt und in die erarbeiteten Modelle eingebunden.

Als Modellierungswerkzeug diente die Modellierungs-Software Rose des Herstellers Rational. Sie ist das Standardwerkzeug für die UML-Modellierung bei IPP.

3.1.5 Entwicklung eines Dokumentationsverfahrens

Zur Entwicklung eines Verfahrens, mit dem UML-Modelle einheitlich online und in gedruckter Form publiziert werden können, wurde eine explorative Arbeitsweise angewandt. Statt einer detaillierten Anforderungs-Analyse im Vorfeld wurde schon frühzeitig ein einfacher Prototyp gefertigt. Dieser diente als Anschauungsmodell, mit dem die Mitarbeiterinnen und Mitarbeiter von IPP die Gelegenheit erhielten, sich ihrer eigenen Vorstellungen über das Publikationssystem bewußt zu werden und diese zu formulieren. Die so erfassten Kommentare und Wünsche wurden dann in einen Anforderungskatalog (s. Abschnitt 5.1) aufgenommen und zur Diskussion gestellt. Dieser Katalog bildete die Grundlage für die schrittweise Erweiterung und Optimierung des Systems.

Für die technische Realisierung des Verfahrens wurden die vielfältigen Möglichkeiten der Sprachfamilie XML (eXtensible Markup Language) genutzt (s. Kapitel 6).

Im Laufe der Entwicklung des Publikationssystems wurde deutlich, dass die zu publizierenden UML-Modelle in einheitlicher Weise erstellt werden sollten. Erst ein definierter Aufbau der Modelle ermöglichte es, die Anforderungen des

Publikationssystems technisch effizient umzusetzen und das System für mehrere Modelle einzusetzen.

Abgesehen von der technischen Seite des Systems erleichtert ein definierter, einheitlicher Modellaufbau das Verständnis der Modellinhalte durch seine Nutzer.

Daher wurden eine Reihe von Modellierungskonventionen entwickelt, die eine einheitliche, konsistente und transparente Anwendung der UML für Geschäftsprozessmodelle bei DESY-IPP sicherstellen sollen (s. Kapitel 4)

3.2 Aufgaben und Prozesse des Anwendersupport

3.2.1 Rolle des Anwendersupports innerhalb von DESY

Inhaltlich ist der IPP-Anwendersupport zwischen Systemtechnik und Fachaufgaben angesiedelt und erfordert oft tiefgehende Kenntnis beider Seiten.

Er unterstützt Systemanwender bei der Bewältigung ihrer Fachaufgaben mit CAD- und Informationssystemen, welche im Rahmen des DESY-Anlagenbaus eingesetzt werden. Im Einzelnen besteht diese Unterstützung aus

- der Schulung von Anwendern durch eigene Veranstaltungen und individuelles Training;
- dem Betrieb einer Anwender-Hotline;
- der technischen Einrichtung und Betreuung von CAD- und Informationssystemen.

IPP bündelt dabei fachliche Kompetenzen aus dem Bereich des Anlagenbaus und der Konstruktion mit dem technischen Know-How über die verwendeten CAD- und Informationssysteme.

Zunächst konzentrierte sich der IPP-Anwendersupport ausschließlich auf die Betreuung von CAD-Systemen bei DESY. Verschiedene Konstruktionsteams bei DESY nutzen CAD, um Komponenten neuer Beschleuniger-Anlagen zu entwerfen oder Bauvorhaben vorzubereiten. Insgesamt werden zur Zeit ca. 300 CAD-Arbeitsplätze von IPP unterstützt. Ca. 200 davon betreffen die Anwendung AutoCAD, ein mid-range 2D-CAD System von der Firma AutoDesk.

Rund 90 Arbeitsplätze bestehen für das System I-DEAS, das high-end 3D-Modellierungswerkzeug des Herstellers EDS. Diese Systeme werden ergänzt durch die gemeinsame Normteile-Bibliothek CADBAS.

Neben den CAD-Anwendungen werden inzwischen weitere Systeme unterstützt, die die verschiedenen Aufgabengebiete des Anlagenbaus unterstützen, unter anderem

- ein Engineering Data Management System (EDMS) für die Unterstützung der Anlagenkonstruktion und -entwicklung, das gleichzeitig als Dokumentenmanagementsystem verwendet wird,
- ein Asset Management System (AMS) für die Unterstützung von Gerätefertigung, -betrieb und -bewirtschaftung,
- ein Geographisches Informationssystem und Facility Management System (GIS/FMS), welches die Installation und den Betrieb gesamter Anlagen unterstützt,

Die Leistungen des IPP-Anwendersupports lassen sich durch fünf Prozesse beschreiben. Dazu gibt das folgende UML-Anwendungsfalldiagramm einen Überblick.

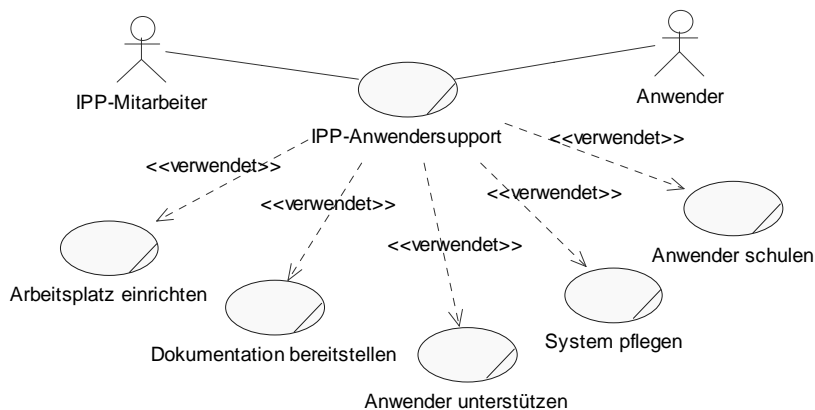


Abbildung 17: Geschäftsprozesse des IPP-Anwendersupport

Ziele, Abläufe und Teilaufgaben der einzelnen Prozesse werden im Folgenden kurz vorgestellt:

3.2.2 Prozess „Arbeitsplatz einrichten“

Dieser Prozess hat zum Ziel, Anwendern schnellstmöglich einen betriebsbereiten Arbeitsplatz mit der bestellten Anwendung zur Verfügung zu stellen.

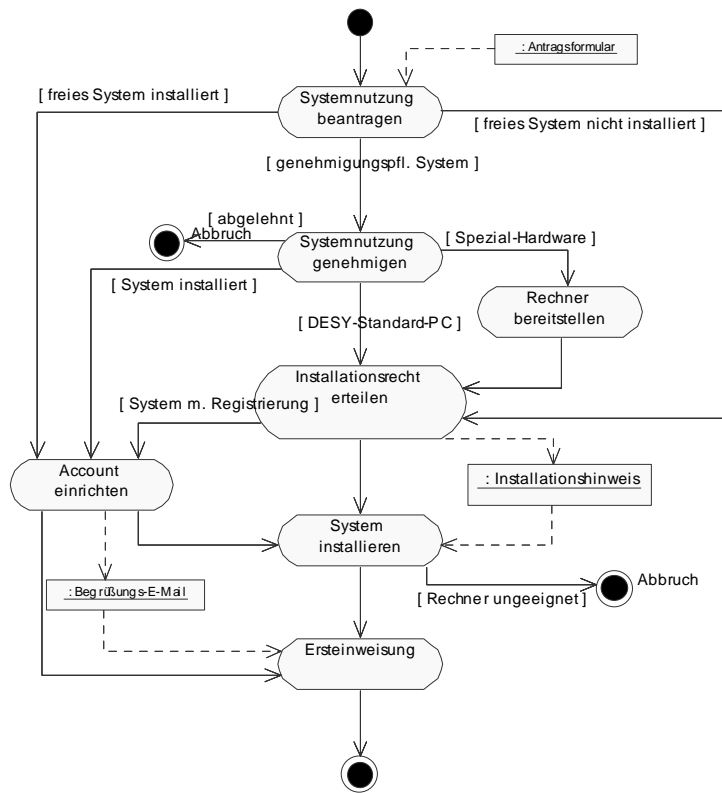


Abbildung 18: Prozessablauf „Arbeitsplatz einrichten“

Beschreibung der dargestellten Aufgaben:

Systemnutzung beantragen

Möchte ein Anwender ein System nutzen, muss er dies beantragen. Dies kann telefonisch, per E-Mail oder mit Hilfe eines Antragsformulars erfolgen.

Systemnutzung genehmigen

Bei genehmigungspflichtigen Systemen wird die beantragte Systemnutzung im Hinblick auf seine Notwendigkeit und die vorhandenen Rahmenbedingungen (Kompetenzen des Anwenders, Kontext der Nutzung etc.) untersucht. Sind die Voraussetzungen erfüllt, wird die Systemnutzung genehmigt.

Rechner bereitstellen

Wenn die Hardware eines DESY-Mitarbeiters nicht ausreicht, um damit ein System zu nutzen, wird ein neuer Rechner mit geeigneter Hardware bereitgestellt. In der Regel geschieht dies durch einen IT-Mitarbeiter.

Installationsrecht erteilen

Wenn alle technischen Voraussetzungen gegeben sind und, bei einem genehmigungspflichtigen System, die Systemnutzung genehmigt wurde, wird dem Anwender durch den IPP-Anwendersupport das Installationsrecht erteilt.

Account einrichten

Einige Systeme erfordern eine Benutzererkennung. Ein entsprechender Account wird nach Erteilung des Installationsrechts eingerichtet, noch bevor das System installiert wird.

System installieren

Je nachdem, ob es über NetInstall verfügbar ist, wird das System durch den Anwender oder einen Mitarbeiter des IPP-Anwendersupports auf dem Rechner des Anwenders installiert.

Ersteinweisung

Nach der Installation eines neuen Systems erhält der Anwender eine Ersteinweisung in die Bedienung und Funktionalität. Diese Ersteinweisung erfolgt entweder persönlich durch einen Mitarbeiter des IPP-Anwendersupports oder mit Hilfe einer automatisierten Unterrichtung per E-Mail.

3.2.3 Prozess „Anwender schulen“

Dieser Prozess hat zum Ziel, Anwender kurzfristig in der Nutzung eines Systems auf dem für sie notwendigen (d.h. prozessbezogenen) Expertise-Level auszubilden.

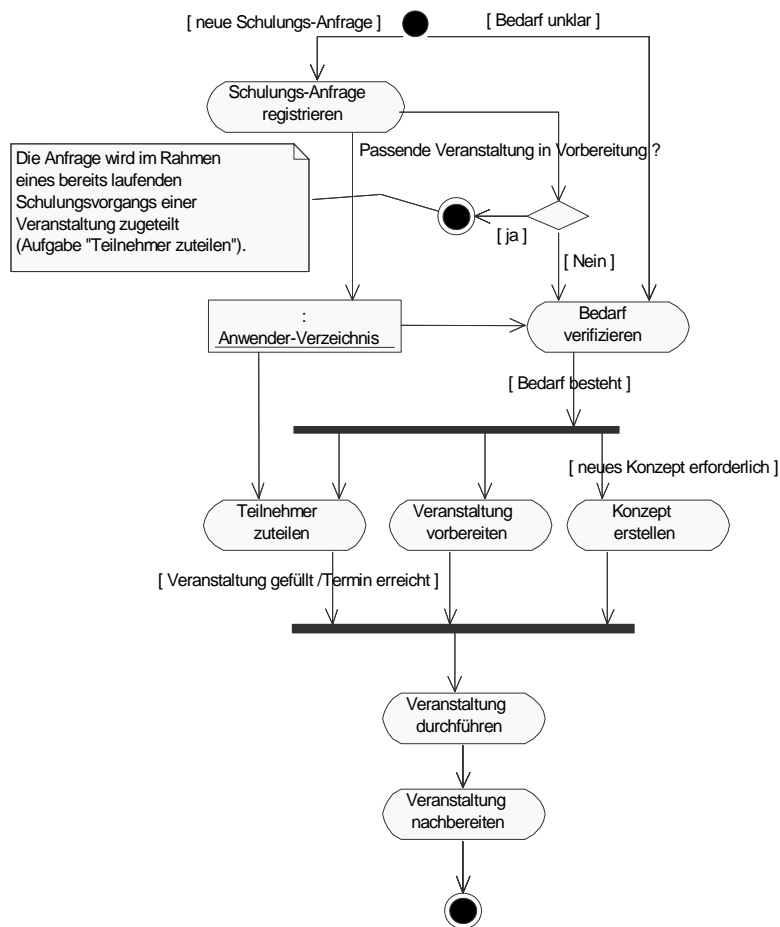


Abbildung 19: Prozessablauf „Anwender schulen“

Beschreibung der dargestellten Aufgaben:

Schulungs-Anfrage registrieren

Eine Anfrage an IPP, die Hinweise auf einen Schulungs-Bedarf beinhaltet, wird als Schulungs-Anfrage im Anwender-Verzeichnis registriert. Dabei werden die inhaltlichen, organisatorischen und didaktischen Wünsche des potentiellen Schulungs- Teilnehmers eingetragen.

Bedarf verifizieren

Dies meint die Aus- und Bewertung der vorliegenden Hinweise über den vorhandenen Schulungsbedarf. Das Ergebnis ist eine Aussage darüber, ob eine Veranstaltung angeboten werden wird.

Teilnehmer zuteilen

Einer Veranstaltung in Vorbereitung werden diejenigen Teilnehmer zugeteilt, die eine Anfrage mit dem entsprechenden Schulungsbedarf gestellt haben.

Veranstaltung vorbereiten

Vor einer Veranstaltung werden Termin und Infrastruktur abgestimmt, Einladungen versendet sowie Lehrmaterialien für die Teilnehmer bereitgestellt.

Konzept erstellen

Ein Konzept beschreibt die inhaltliche und didaktische Durchführung einer Veranstaltung. Dabei verbindet es die Umsetzung eines oder mehrerer Schulungstypen mit der Vermittlung eines oder mehrerer Themen.

Veranstaltung durchführen

Diese Aufgabe umfasst die Tätigkeiten in unmittelbarem zeitlichen Zusammenhang mit der Schulungsveranstaltung selbst.

Veranstaltung nachbereiten

Diese Aufgabe beinhaltet die Auswertung der Veranstaltung, die Durchführung und Buchung von Zahlungen, die Ergänzung der Statistik sowie die Ausstellung der Zertifikate für die Teilnehmer.

3.2.4 Prozess „Anwender unterstützen“

Dieser Prozess hat zum Ziel, Anfragen von Anwendern in schnellst möglicher, vorab vereinbarter Zeit zur Zufriedenheit der Anwender zu beantworten.

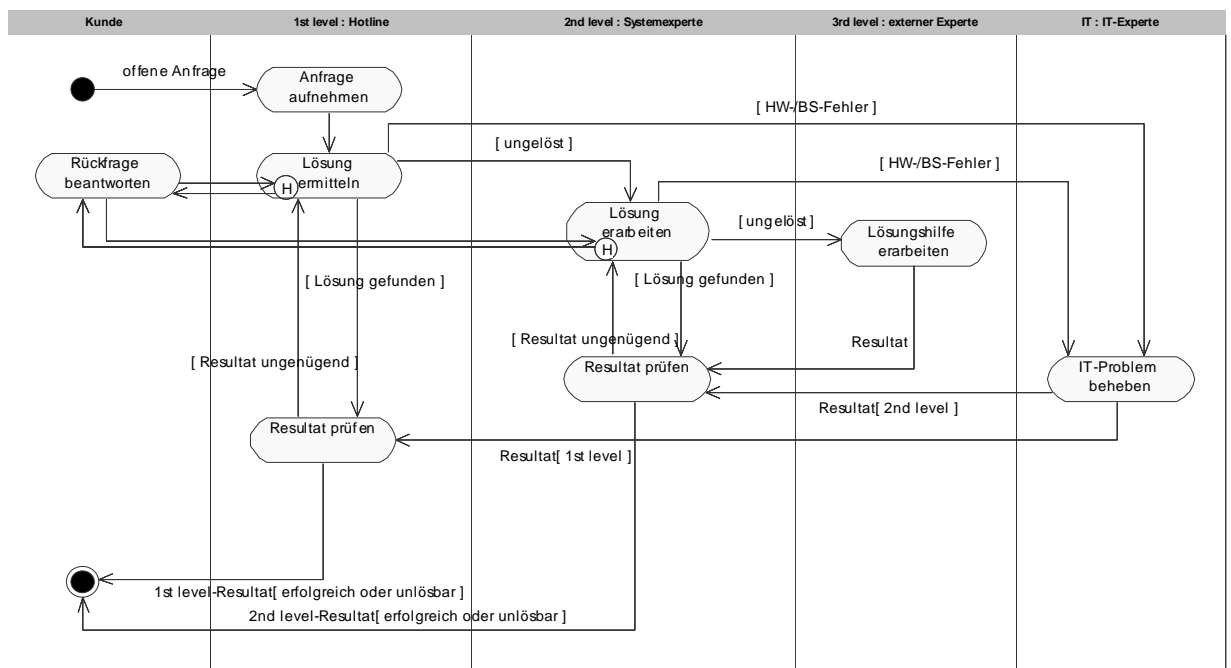


Abbildung 20: Prozessablauf „Anwender unterstützen“

Beschreibung der dargestellten Aufgaben:

Anfrage aufnehmen

Eine Anfrage wird immer vom 1st-level-support aufgenommen. Sie trifft telefonisch oder per E-Mail ein. Die Anfrage wird kategorisiert und die wichtigsten Daten werden erfasst.

Lösung ermitteln

Nach Aufnahme der Anfrage beginnt die Ermittlung einer Lösung durch den 1st-level-Mitarbeiter. Stellt er einen Hardware- oder Betriebssystemfehler fest, leitet er die Anfrage an die IT-Abteilung weiter. Findet er selbst eine Lösung, prüft er das Resultat. Bleibt die Anfrage ungelöst, beauftragt er einen Systemexperten, eine Lösung zu erarbeiten.

Rückfrage beantworten

Eine Rückfrage entsteht während der Ermittlung oder Erarbeitung einer Lösung und wird vom Kunden beantwortet.

Lösung erarbeiten

Stellt der Systemexperte fest, dass es sich um einen Hardware- oder Betriebssystemfehler handelt, leitet er die Anfrage an einen IT-Experten weiter. Findet er selbst eine Lösung, prüft er das Resultat. Kommt er zu keiner Lösung, zieht er einen externen Experten hinzu, der eine Lösungshilfe erarbeiten soll.

Lösungshilfe erarbeiten

Lösungshilfen werden von externen Experten der Service- bzw. Herstellerfirmen erarbeitet, wenn der Systemexperte bei IPP alleine die Anfrage nicht zu einem erfolgreichen Resultat führen kann. Das Resultat der Lösungshilfe leitet er an den Systemexperten, der es prüft.

IT-Problem beheben

Diese Aufgabe betrifft Hardware- und Betriebssystemfehler, die von der IT-Abteilung behoben werden. Das Resultat wird an die Hotline bzw. den Systemexperten zur Prüfung weitergeleitet.

Resultat prüfen

Hotline-Mitarbeiter bzw. Systemexperten prüfen das Resultat, d.h. die Qualität der von ihnen ermittelten bzw. erarbeiteten Lösungen. Ebenso prüfen sie die Resultate der Lösungshilfen und Problem-Behebungen der externen bzw. IT-Experten. Sind die Resultate ungenügend, werden die Aufgaben 'Lösung ermitteln' bzw. 'Lösung erarbeiten' erneut aktiviert. Ist das Resultat eine erfolgreiche Lösung oder handelt es sich um ein unlösbares Problem, wird der Kunde darüber unterrichtet.

3.2.5 Prozess „Dokumentation bereitstellen“

Dieser Prozess hat zum Ziel, dass Anwender im Intranet schnell und ergonomisch ausreichende Informationen finden, um ihre Anfragen selbständig lösen zu können.

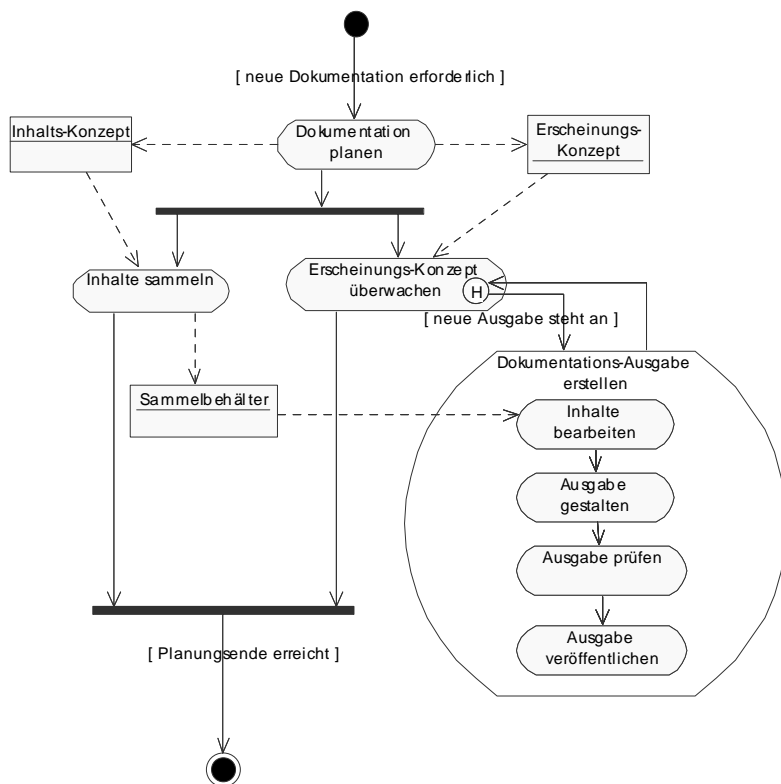


Abbildung 21: Prozessablauf "Dokumentation bereitstellen"

Beschreibung der dargestellten Aufgaben:

Dokumentation planen

Inhalt, Struktur, Erscheinungsweise und Erscheinungsfrequenz werden für eine Dokumentation in Übereinstimmung mit IPP-internen und -externen Anforderungen geplant. Die Planung mündet in ein Inhaltskonzept und ein Erscheinungskonzept.

Inhalte sammeln

Die für eine Dokumentation relevanten Quellen werden systematisch und kontinuierlich ausgewertet. Dabei werden solche Inhalte selektiert, die den Kriterien des Inhaltskonzepts einer Dokumentation entsprechen.

Erscheinungs-Konzept überwachen

Die Termine und Kriterien des Erscheinungskonzepts werden kontinuierlich verfolgt. Immer wenn ein Termin für eine neue Dokumentations-Unterlage erreicht ist oder andere Kriterien für ihre Erstellung erfüllt sind, wird die Aufgabe "Dokumentations- Ausgabe erstellen" ausgeführt.

Dokumentations-Ausgabe erstellen

Bei der Erstellung einer Dokumentations-Ausgabe - gedruckt, online oder in anderer Form - werden zunächst die Inhalte bearbeitet, dann die Ausgabe gestaltet und geprüft. Schließlich wird sie veröffentlicht.

Inhalte bearbeiten

Die Vorgaben des Erscheinungskonzepts einer Dokumentation werden für eine spezielle Ausgabe der Dokumentation umgesetzt. Die gesammelten Inhalte werden dazu gesichtet, strukturiert, komprimiert und ggf. umformuliert und ergänzt.

Ausgabe gestalten

Die fertig bearbeiteten Inhalte einer Dokumentation werden mit einer passenden Formatierung versehen. Für die gesamte Ausgabe wird ein geeignetes Layout entworfen und implementiert.

Ausgabe prüfen

Auf Grundlage sowohl des Inhalts-, als auch Erscheinungskonzepts wird geprüft, ob die gestaltete Ausgabe den dort niedergelegten Kriterien entspricht. Festgestellte Defizite werden gleich behoben. Werden keine Defizite (mehr) festgestellt, wird die Ausgabe zur Veröffentlichung freigegeben.

Ausgabe veröffentlichen

Eine Ausgabe wird gedruckt und verteilt oder/und online im Intranet dem anvisierten Nutzerkreis zur Verfügung gestellt. Ein weiterer möglicher Weg der Veröffentlichung wäre die Verbreitung per E-Mail.

3.2.6 Prozess „System pflegen“

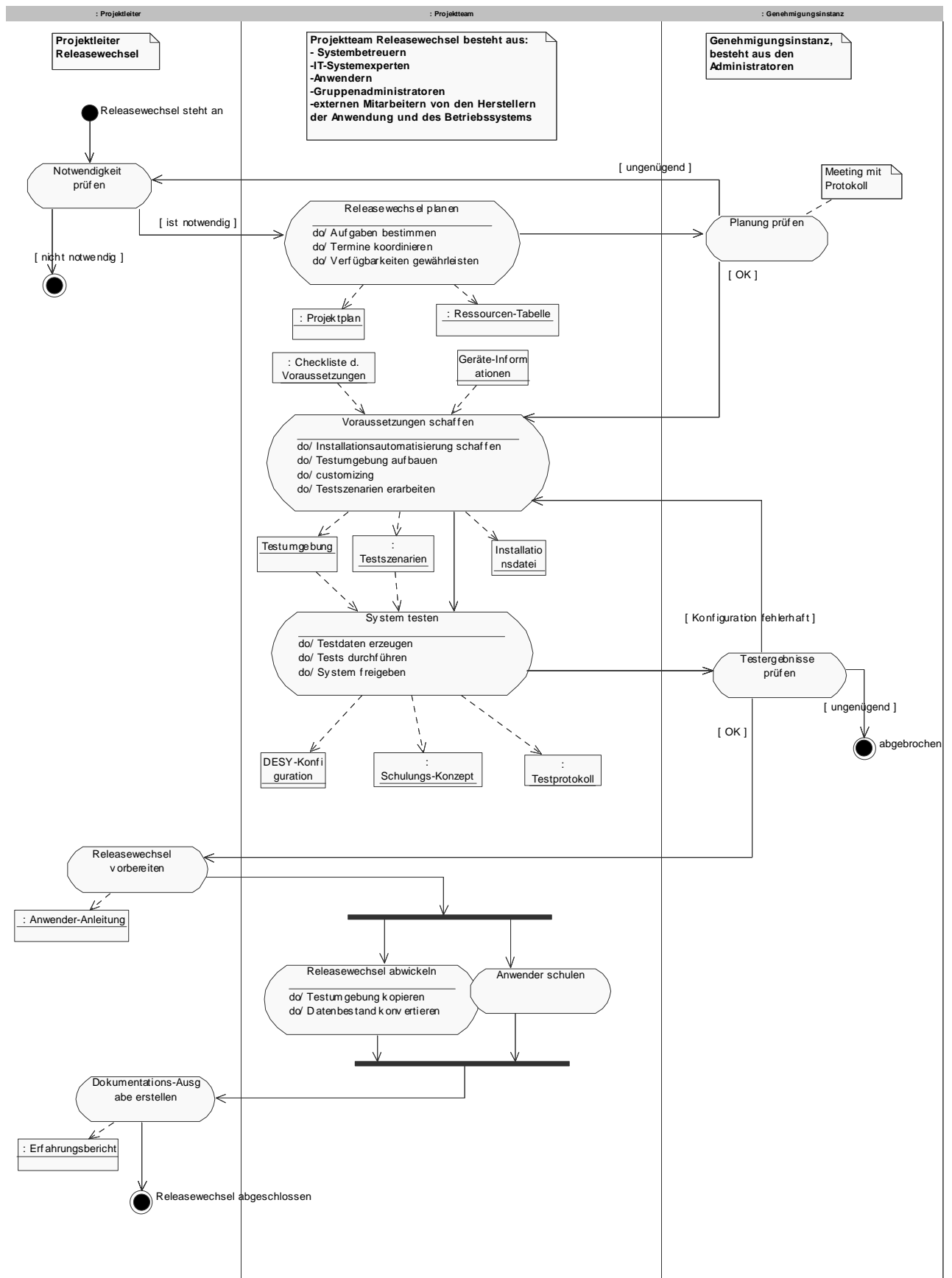


Abbildung 22: Prozessablauf "System pflegen"

Dieser Prozess hat zum Ziel, unterstützte Systeme auf einem technisch hochwertigen Stand zu halten, der den Anforderungen der Anwender genügt.

Beschreibung der dargestellten Aufgaben:

Notwendigkeit prüfen

Die Notwendigkeit eines Releasewechsels wird zu Beginn des Prozesses durch den Projektleiter geprüft.

Releasewechsel planen

Die einzelnen Aufgaben-Schritte werden definiert. Termine werden abgestimmt und die entsprechende Verfügbarkeit der Ressourcen veranlasst. Die Ergebnisse der Aufgabe werden in einem Projektplan und einer Ressourcen-Tabelle zusammengefasst.

Planung prüfen

Bevor der Prozess von der Planungs- in die Testphase übergeht, wird die zuvor erstellte Planung durch die Genehmigungsinstanz geprüft. Bewertet sie die Planung als ungenügend, wird erneut die Notwendigkeit zur Durchführung des Prozesses geprüft.

Voraussetzungen schaffen

Diese Aufgabe schafft die entscheidenden Voraussetzungen für den Test des neuen Release. Neben der Erstellung der Installationroutine und einer Testumgebung werden bestehende Einstellungen des bisherigen Release an das neue Release angepasst ("customizing"). Die für den Test erforderlichen Testszenarien werden ausgearbeitet.

System testen

Der Nutzen und die Zuverlässigkeit eines neuen Release wird anhand von Szenarien getestet. Daraus ergibt sich die DESY-spezifische Konfiguration sowie der Inhalt der Schulungsunterlagen.

Testergebnisse prüfen

Es wird geprüft, ob die getestete Konfiguration fehlerhaft ist und ggf. überarbeitet werden muss. Sind die Testergebnisse ungenügend, führt die Prüfung zum Abbruch des Releasewechsels. Nur bei fehlerfreien Tests wird anschließend der Releasewechsel vorbereitet.

Releasewechsel vorbereiten

Auf der Grundlage der Ergebnisse der Testphase wird eine Anwender-Anleitung erstellt und verteilt, anhand derer die Anwender die Installation durchführen können.

Releasewechsel abwickeln

Die Installations-Routine wird allen Anwendern zur Verfügung gestellt, die dann in einem festgelegten Zeitraum das System auf ihrem eigenen Rechner installieren. Außerdem wird der Datenbestand des alten Release auf das neue Release konvertiert.

Dokumentations-Ausgabe erstellen

Bei der Erstellung einer Dokumentations-Ausgabe - gedruckt, online oder in anderer Form - werden zunächst die Inhalte bearbeitet, dann die Ausgabe gestaltet und geprüft. Schließlich wird sie veröffentlicht.

Anwender schulen

Dieser Prozess hat zum Ziel, Anwender kurzfristig in der Nutzung eines Systems auf dem für sie notwendigen (d.h. prozessbezogenen) Expertise-Level auszubilden (für Details s. Abschnitt 3.2.2)

3.2.7 Zentrales Arbeitsmittel: Der Request-Tracker

Die vorgestellten Prozesse geben vor, wie einzelne Vorgänge in der Regel bearbeitet werden. Auch wenn somit die Vorgänge vereinheitlicht und im Grundsatz nachvollziehbar gestaltet werden, ist es darüber hinaus sinnvoll, die Daten der einzelnen Vorgänge zu dokumentieren. Zu diesem Zweck wurde bei IPP das Anfragedokumentationssystem *Request Tracker* eingeführt. Es handelt sich dabei um ein Informationssystem zur Dokumentation und Verwaltung von Anfragen und Geschäftsvorfällen in einer Organisation. Er basiert auf dem Prinzip, dass jeder Vorgang einschließlich seiner Bearbeitungsstadien und Statusänderungen erfasst wird. Ein Vorgang wird dabei durch genau ein Ticket repräsentiert, Bearbeitungsschritte werden durch Kommentare erfasst und in der Ticket-Historie erfasst. Das folgende UML-Klassendiagramm zeigt die Objektstruktur des *Request Trackers*.

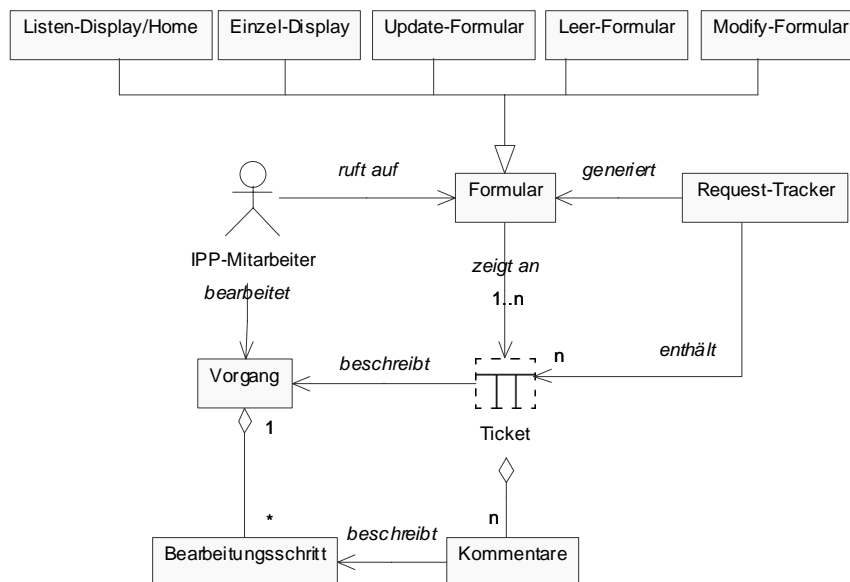


Abbildung 23: Objektstruktur des Request-Trackers bei IPP

3.3 Anforderungen der Prozessdokumentation

IPP ist auf dem Weg, eine prozessorientierte Arbeitsweise in allen Bereichen des Anwendersupports konsequent umzusetzen. Daher sollen einerseits die entworfenen Prozesse stärker in den Arbeitsalltag der Mitarbeiter integriert werden. Zum Anderen gehört dazu die regelmäßige Überarbeitung der bestehenden Prozesse mit dem Ziel, sie optimal an sich wandelnde Anforderungen anzupassen.

Für diesen Zweck sind dokumentierte Prozessmodelle notwendig, die als Arbeits- und Diskussionsgrundlage dienen. Um die Prozessmodelle effizient aufbereiten und kommunizieren zu können, müssen Voraussetzungen auf zwei Ebenen geschaffen werden.

1. Die Prozessmodelle sollten nach einem einheitlichen Regelwerk (Modellierungskonventionen) erstellt werden, das die Verständlichkeit sowie die konsistente Erstellung und Änderung erleichtert. Die Konventionen sollen zudem folgende Anforderungen für die praktische Verwendung der Modelle berücksichtigen:
 - a. Bereitstellung von Handlungsanweisungen für die Ausführung der Prozessschritte;
 - b. Einbindung von Arbeits- und Hilfsmitteln;
 - c. Definition von Tätigkeitsprofilen der Akteure;
 - d. Einheitliche Strukturierung verschiedener Modelle für ein effizientes Publikationssystem (s. Punkt 2).
2. Die Dokumentation der Prozessmodelle sollte durch ein Publikationssystem unterstützt werden, das die Modellinhalte in zielgruppengerechter Weise sowohl in interaktiv navigierbarer Form, als auch mit Hilfe von ausdruckbaren Versionen über das Intranet kommuniziert. Die folgenden Anforderungen sind dabei wesentlich:
 - a. Darstellung der Handlungsanweisungen;
 - b. Zugriff auf die Arbeits- und Hilfsmittel;
 - c. Möglichkeit zur Ansicht von Tätigkeitsprofilen;
 - d. Möglichkeit zur effizienten Aktualisierung der Modellpublikationen;
 - e. Parallele Publikation mehrerer Modelle;

Die konkrete Umsetzung dieser Voraussetzungen und Anforderungen wird in den nachfolgenden Kapiteln erläutert.

4 Entwicklung spezifischer Modellierungskonventionen

Mit den im Folgenden entwickelten Modellierungskonventionen sollen für die spezifischen Anforderungen der Prozessdokumentation (s. Kapitel 3) geeignete Lösungen festgelegt werden. Sie sind eine wesentliche Grundlage für die Funktionalität und den technischen Aufbau des Publikationssystems.

Außerdem macht es die große Flexibilität und Variationsbreite der UML erforderlich, dass ihre spezifische Anwendung an den jeweiligen Kontext und Zweck der Modelle angepasst wird. Damit soll ein einheitliches und konsistentes Vorgehen bei der Modellierung erreicht werden.

Eine einheitliche Verwendung der UML hat dabei einen mehrfachen Nutzen:

- Die Verständigung der an der Modellierung beteiligten Personen wird vereinfacht.
- Die Kommunikation der Modelle gegenüber der betroffenen Mitarbeiterschaft wird erleichtert.
- Die Standards dienen als Anleitung und Referenz für die zügige Durchführung von Änderungen und Ergänzungen an den Modellen sowie für die effiziente Erstellung neuer Modelle.
- Besonderheiten des Modellierungswerkzeugs und des Publikationssystems können von vornherein berücksichtigt werden. Spielräume können genutzt und Einschränkungen umgangen bzw. ausgeglichen werden.

Gegenstand der Modellierungskonventionen sind die im Rahmen der gewählten Modellierungssprache UML und des Modellierungswerkzeugs *Rational Rose* umgesetzten Modellierungstechniken sowie die verwendeten Elementtypen und Beziehungstypen (vgl. dazu Becker 2000, S. 68ff.).

Im Folgenden werden die wesentlichen Bereiche der Standards erarbeitet und dabei konkrete Konventionsvorschläge vorgestellt. Dabei wird kein Anspruch auf vollständige Erfassung aller Fragestellungen der Prozessmodellierung erhoben. Vielmehr besteht in den nicht geregelten Bereichen Spielraum für modellspezifische Lösungen.

4.1 Aufbau und Gliederung eines Modells

Wie in Abschnitt 2.3 beschrieben, kann ein Geschäftsmodell durch unterschiedliche Sichten beschrieben werden. Als gemeinsamer Nenner der meisten Sichtenkonzepte wurden dabei die Funktionssicht, die Objektsicht, die Organisationssicht und die Prozesssicht ausgewählt.

Für eine effiziente und einheitliche Vorgehensweise bei der Modellierung ist es hilfreich, eine Sicht als primäre Sicht zu definieren, die für das gesamte Modell ein Grundgerüst liefert, auf dem die anderen Sichten aufbauen können. Diese primäre Sicht kann gleichzeitig den Nutzern des Publikationssystem als Orientierung innerhalb des Modells dienen.

Die Aufbaustruktur als bestimmende Dimension einer Organisation zu betrachten und damit die Organisationssicht als primäre Sicht zu definieren, entspricht nicht (mehr) dem aktuellen Diskussionsstand der Betriebswirtschaft (s. Kapitel 1).

Im Sinne einer prozessorientierten Modellierung liegt es nahe, die Prozesssicht als

primäre Sicht zu wählen. Dies ist jedoch nicht praktikabel, da in Abläufen zwei Aussagen über das Modell zugleich getroffen werden: Die Auswahl der Aufgaben und die Art ihrer Verkettung. Beides gleich zu Beginn der Modellierung festlegen zu sollen, stellt eine hohe Hürde und Anforderung dar. Zudem stellt ein Ablaufdiagramm nur die Aufgaben und den Ablauf innerhalb eines einzelnen Prozesses dar, kann jedoch nur schlecht einen Überblick über das gesamte Aufgabenspektrum der Organisation geben. Dieses kann jedoch innerhalb der Funktionssicht geleistet werden. Abbildung 17 auf Seite 50 zeigt die entsprechende Aufgabenstruktur des Anwendersupports bei IPP.

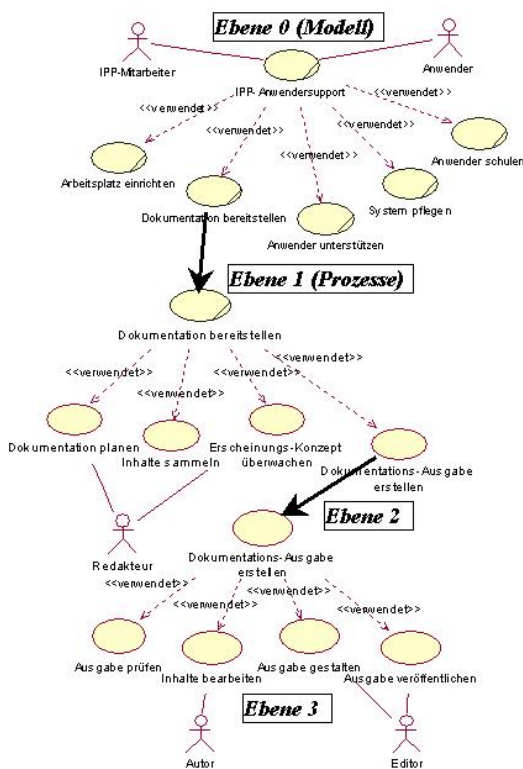


Abbildung 24: Hierarchische Aufgabenstruktur mit Anwendungsfällen

Aufgaben können in Teilaufgaben zerlegt werden, die wiederum in kleinere Teilaufgaben zerlegt werden können bis hin zu der Definition elementarer Aktionen. Die Basisstruktur eines Modells kann also als ein hierarchisch gegliederter Baum von Prozessen und Aufgaben betrachtet werden, dessen Wurzel das gesamte Modell selbst repräsentiert (s. Abbildung 24).

In der UML werden Aufgaben durch Anwendungsfälle dargestellt. Um ganze Prozesse von Aufgaben innerhalb eines Prozesses unterscheiden zu können, kann der Stereotyp „Geschäftsanwendungsfall“ (Business Use Case) verwendet werden.

Ausgehend von der Basisstruktur kann nun jedes Diagramm im Modell einer bestimmten Aufgabe zugeordnet werden. *Rational Rose* unterstützt diese Form der Gliederung, indem Diagramme im Repository auch unter Anwendungsfällen abgelegt werden können. Dabei muss allerdings auch beachtet werden, dass ein Diagramm Sachverhalte beinhalten kann, die für mehrere Aufgaben relevant sind.

Da Anwendungsfälle in *Rational Rose* zwar Diagramme enthalten können, aber keine anderen Anwendungsfälle, ist es im Sinne einer systematischen Ablage und Verwaltung der Funktionen ratsam, Packages (s. Glossar) zur Gliederung zu verwenden.

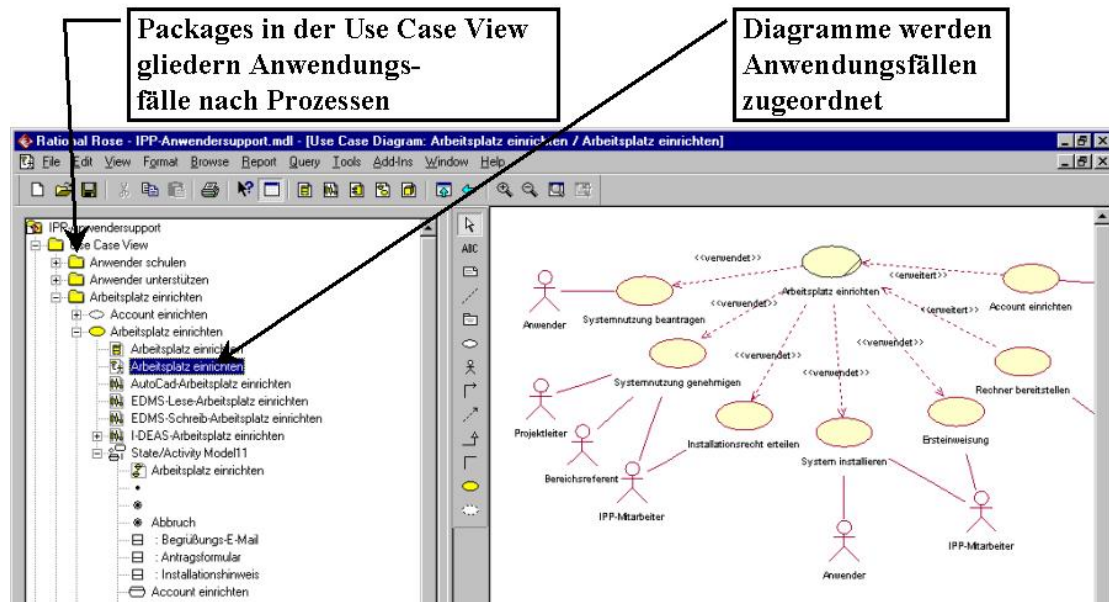


Abbildung 25: Gliederung von Anwendungsfällen und Diagrammen in *Rational Rose*

Konventionsvorschläge:

- i. Das Grundgerüst eines Prozessmodells besteht aus einer hierarchischen Struktur von Aufgaben, die durch Anwendungsfälle repräsentiert werden.*
- ii. Das Modell selbst steht an der Spitze der Hierarchie und wird durch einen Anwendungsfall des Stereotyps „business use case“ repräsentiert (Ebene 0).*
- iii. Die Prozesse werden durch jeweils einen Anwendungsfall des Stereotyps „business use case“ repräsentiert (Ebene 1).*
- iv. Aufgaben, die Teil eines Prozesses oder einer anderen Aufgabe sind, werden durch einfache Anwendungsfälle (ohne Stereotyp) dargestellt (Ebenen 2-N).*
- v. Modell, Prozesse und Aufgaben werden durch Anwendungsfall-, Aktivitäts-, Sequenz- und Klassendiagramme näher beschrieben, die dem jeweiligen Anwendungsfall zugeordnet werden.*
- vi. Für jeden Prozess wird ein gleichnamiges Package in der „Use Case View“ angelegt. Die Anwendungsfälle (Prozess incl. Aufgaben und Teilaufgaben) eines Prozesses werden in das entsprechende Package gelegt. Der Anwendungsfall der Ebene 0 wird direkt in der „Use Case View“ angelegt.*

4.2 Funktionssicht

4.2.1 Darstellung der Aufgabenhierarchie

Die hierarchische Anordnung der Aufgaben beruht auf einer Teil-Ganzes-Beziehung zwischen einer Aufgabe und ihren Teilaufgaben. Dabei soll zwischen obligatorischen und optionalen Teilaufgaben unterschieden werden. UML-Anwendungsfalldiagramme bilden diese Zusammenhänge über die Beziehungen „erweitert“ und „verwendet“ ab (s. Abbildung 26). Sie sollten der Aufgabe zugeordnet werden, deren Teilaufgaben sie zeigen.

Komplexe Modelle besitzen häufig mehr als zwei hierarchische Ebenen innerhalb ihrer Aufgabenstruktur. Für eine übersichtliche Darstellung ist es sinnvoll, auf einem Diagramm nicht mehr als eine Teilaufgaben-Ebene unterhalb der betrachteten Aufgabe darzustellen. Für jede Ebene innerhalb eines Astes des Hierarchiebaumes muss demnach ein weiteres Anwendungsfalldiagramm erstellt werden.

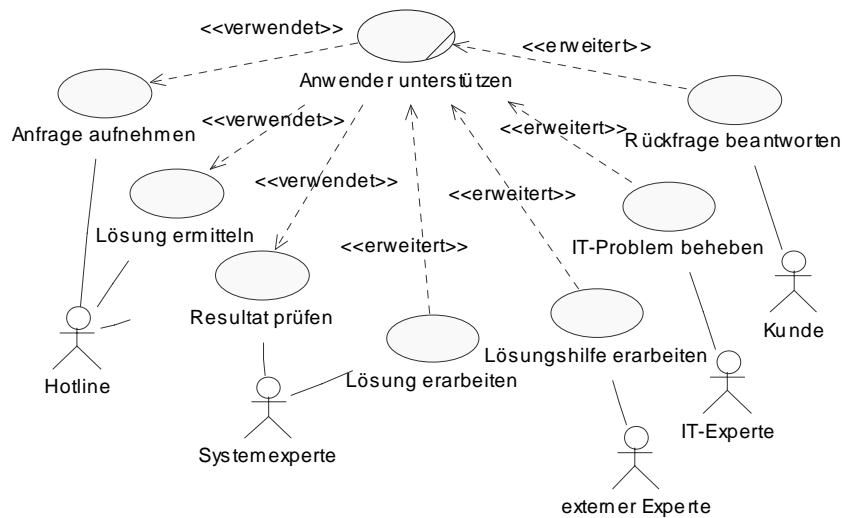


Abbildung 26: Aufgabe mit obligatorischen und optionalen Teilaufgaben

Konventionsvorschläge

- vii. Die Teilaufgabenstruktur einer Aufgabe wird durch ein Anwendungsfalldiagramm dargestellt. Dazu wird dem Anwendungsfall einer Aufgabe mit Teilaufgaben genau ein Anwendungsfalldiagramm zugeordnet, das die Relationen zwischen Aufgabe (Ebene N) und deren Teilaufgaben (Ebene N+1) zeigt.
- viii. Zwischen Anwendungsfällen werden nur die Relationen „verwendet“ oder „erweitert“ benutzt. Ein Anwendungsfall „verwendet“ andere Anwendungsfälle (bei obligatorischen Teilaufgaben) oder wird durch diese „erweitert“ (bei optionalen Teilaufgaben).

4.2.2 Mehrfachverwendung von Aufgaben in verschiedenen Kontexten

Die Aufgabenstruktur eines Geschäftsprozessmodells zeigt, dass unterschiedliche Geschäftsprozesse einer Organisation gemeinsame Berührungspunkte haben und sich teilweise auch überschneiden.

So sind Aufgaben denkbar, die in mehreren Geschäftsprozessen verwendet werden. Auch ganze Geschäftsprozesse können innerhalb bzw. durch andere Geschäftsprozesse ausgelöst werden.

Bei IPP ist eine derartige Verknüpfung im Prozess „System pflegen“ zu erkennen (s. Abschnitt 3.2.6). Im Rahmen eines Release-Wechsels (entspricht einem Vorgang des Prozesses „System pflegen“) wird sowohl der Prozess „Anwender schulen“ als auch die Aufgabe „Dokumentationsausgabe erstellen“ verwendet, welche auch Teil des Prozesses „Dokumentation bereitstellen“ ist.

Ein Problem entsteht dabei aus dem Bedarf der Darstellung des Prozessmodells in einer Print-Dokumentation. Während für die Hyperlink-Struktur einer Online-

Publikation die Reihenfolge der Beschreibung von Prozesselementen nachrangig ist, ist diese für die Verständlichkeit einer gedruckten Dokumentation essentiell. Auch das Problem doppelter Beschreibungen, die bei der mehrfachen Verwendung von Aufgaben an mehreren Stellen des Prozesses entstehen, muss mit dem Anspruch einer übersichtlichen Darstellung in Einklang gebracht werden.

Eine Lösung besteht darin, eine Aufgabe auch bei einer Mehrfachverwendung in mehreren Prozessen immer einem Prozess primär zuzuordnen. In einem Prozess wird also generell zwischen eigenen und fremden Aufgaben unterschieden. Eine Aufgabe wird bei Verwendung in anderen Prozessen diesen nur „geliehen“. In der Print-Dokumentation erfolgt dann die Beschreibung der eigenen Aufgaben vor der Beschreibung der geliehenen Aufgaben. Auch kann bei fremden Aufgaben statt einer Wiederholung aller einzelnen Details der Beschreibung ein Verweis auf die Beschreibung des Prozesses stehen, dem die Aufgabe entliehen wurde.

Die Bedeutung der Aufgabe kann häufig am Besten über die Funktion der Aufgabe innerhalb eines Prozessablaufs beschrieben werden. Je deutlicher die Aufgabe aber eine prozessübergreifende Rolle spielt, um so allgemeiner und abstrakter muss sie definiert sein, um keine logischen Widersprüche bzw. Inkonsistenzen hervorzurufen. Dies geht zu Lasten der Verständlichkeit für den Nutzer des Prozessmodells, der eine möglichst anschauliche Beschreibung der Vorgänge benötigt. Aufgaben sollten daher nur in geringem Umfang in mehreren Prozessen verwendet werden. Und mit der primären Zuordnung einer Aufgabe zu einem Prozess sollte sich auch die Beschreibung vor allem an der Verwendung in diesem Prozess orientieren.

Um die primäre Zuordnung einer Aufgabe zu einem Prozess zu ermöglichen, muss ein geeignetes Kennzeichen gefunden werden. Eine Möglichkeit ist die Definition eines zusätzlichen Eigenschaftsfeldes („tagged values“) für Anwendungsfälle. *Rational Rose* bietet bereits ein Feld „rank“ als Standarderweiterung an, mit dem zudem die Reihenfolge in der Dokumentation gesteuert werden kann.

Konventionsvorschläge:

- ix. *Eine Aufgabe wird einem Prozess primär zugeordnet. Die Zuordnung wird in den Rängen der Anwendungsfälle definiert (s. Konvention xl).*
- x. *Aufgaben werden in geeigneten Fällen auch in anderen als dem primär zugeordneten Prozess verwendet. Im Zweifelsfall wird ein neuer Anwendungsfall definiert.*

4.3 Organisationssicht

4.3.1 Darstellung der Akteure

Die Ausrichtung der personellen Organisationsressourcen an Geschäftsprozessen lässt eine eigenständige Sicht auf den inneren statischen Aufbau einer Organisation in den Hintergrund treten. Von deutlich relevanterer Bedeutung ist die Verknüpfung zwischen Akteuren auf der einen Seite und Aufgaben, Abläufen und Objekten auf der anderen Seite. Der Klassen-Stereotyp „Actor“ innerhalb der UML kann zur Darstellung dieser Verknüpfung flexibel eingesetzt werden. In Klassendiagrammen können die Beziehungen der Akteure untereinander und zu den relevanten Objektklassen eines Prozesses dargestellt werden. Akteure auf Anwendungsfalldiagrammen können zudem die Verantwortungsbereiche der Mitarbeiter transparent machen.

Schwimmbahnen in Aktivitätsdiagrammen und aus „Actor“-Klassen abgeleitete Akteure in Sequenzdiagrammen stellen die Verknüpfung zur Prozesssicht her.

Konventionsvorschläge:

- xi. Akteure der Prozesse werden durch Klassen des Stereotyps „Actor“ repräsentiert und in einem Package in der Logical View abgelegt.*
- xii. Für die Darstellung aller Akteure eines Modells einschließlich ihrer Beziehungen untereinander wird ein eigenständiges Klassendiagramm erstellt, das dem Anwendungsfall der Ebene 0 zugeordnet wird.*
- xiii. Diejenigen Akteure, die für eine dargestellte Aufgabe eine wesentliche Durchführungsverantwortung wahrnehmen, werden auf den Anwendungsfalldiagrammen über ungerichtete Assoziationen mit den jeweiligen Anwendungsfällen verbunden.*

4.3.2 Darstellung von Tätigkeitsprofilen

Prozessmodelle dienen u.a. dazu, den beteiligten Mitarbeitern die an sie gestellten Anforderungen vor Augen zu führen.

Um grobe Zuständigkeiten zu verdeutlichen, genügt es, die Akteure konsequent mit ihren eigenen Aufgaben zu verknüpfen. Dies kann auf Anwendungsfall-Diagrammen der UML geschehen, die die Aufgabenstruktur mit den beteiligten Akteuren zeigen (s. Abbildung 26).

Um Tätigkeiten aber im Detail zu bestimmen, sollten geeignete Methoden für die Akteure definiert werden. Die Gesamtheit der zugehörigen Methoden aus den

verschiedenen Prozessen kann somit das Tätigkeitsprofil eines Akteurs widerspiegeln.

Im Gegensatz zu (technischen) Geschäftsobjekten, deren Methoden sehr spezifische und klar abgegrenzte Funktionen beinhalten, sind die Anforderungen an das Personal einer Organisation weitergehend. Es entspricht modernen und bewährten Auffassungen über Personalführung, dass Mitarbeiter nicht nur Kenntnisse über genau ihre eigenen Aufgaben haben, sondern auch über die Aufgaben der anderen Mitarbeiter ihrer Organisation. Dies ermöglicht es u.a., dass Aufgaben rotieren können, Vertretungen einfacher zu organisieren sind und überhaupt der Personaleinsatz flexibler geplant werden kann.

Dieses Ziel führt dazu, die für die Geschäftsprozesse erforderlichen Kenntnisse unabhängig von den einzelnen Akteuren zu betrachten.

In der UML kann die Trennung von Akteuren und Kenntnissen durch die Einführung von Kenntnisklassen erfolgen, die die Methoden jeweils eines Anforderungsbereichs umfassen. Sie vererben ihre Methoden an diejenigen Akteure, die ihrem Anforderungsbereich zugeordnet werden. Die folgende Abbildung zeigt diesen Zusammenhang bei IPP.

Bei der Darstellung von Tätigkeitsprofilen mittels definierter Methoden muss beachtet werden, dass der Nutzer eines Modells in der Regel Vollständigkeit erwartet. Wenn das Modell die Anforderungen an das Personal also umfassend darstellen soll, müssen auch die definierten Methoden entsprechend umfassend alle Bereiche abdecken. Abbildung 27 ist in dieser Hinsicht noch nicht vollständig.

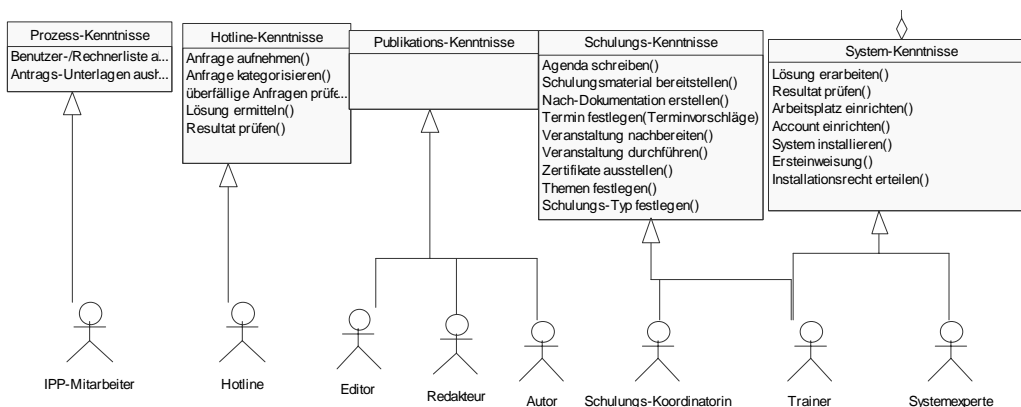


Abbildung 27: Separate Definition von Kenntnisklassen zur Erstellung von Tätigkeitsprofilen

Konventionsvorschläge:

- xiv. *Akteure erhalten keine eigenen Methoden, sondern die jeweiligen Klassen erben Methoden aus separaten Kenntnisklassen, die Cluster von Tätigkeiten repräsentieren.*
- xv. *Für ein Tätigkeitsprofil wesentliche Methoden werden in den Kenntnisklassen auch dann angelegt, wenn sie in keinem Szenario aufgerufen werden.*

4.4 Objektsicht

4.4.1 Darstellung von Objekten

Ähnlich wie bei der Organisationssicht, hat auch bei der Objektsicht die Verknüpfung zur Funktionssicht und Prozesssicht eine wesentliche Bedeutung. Darüberhinaus erfüllt die Objektsicht aber weitergehende Anforderungen für die Modellkonsistenz und die korrekte Abbildung der Realität. So wird mit ihr die Begriffswelt der Objekte festgelegt, die in der betrieblichen Realität verwendet oder produziert werden. Außerdem dient sie dazu, die Schnittstellen zu den beteiligten Informationssysteme zu zeigen und somit für Wartungen und Erweiterungen leichter verständlich zu machen. Je nach Detaillierungsgrad des Modells werden sehr viele Objektklassen für eine vollständige Beschreibung benötigt, so dass ihre Zahl diejenige der Aufgaben meist deutlich übersteigt. Viele Objektklassen werden in mehreren Prozessen des Modells verwendet und können daher nicht einer Aufgabe eindeutig zugeordnet werden. Daher bietet sich eine übersichtliche Gliederung der Klassen in *Rational Rose* in der Logical View unabhängig von den Anwendungsfällen an.

Eine übersichtliche und konsistente Dokumentation macht es zudem erforderlich, Klassen nach gemeinsamen Merkmalen zu gruppieren und die Klassendiagramme den Anwendungsfällen zuzuordnen, die den passenden Kontext liefern.

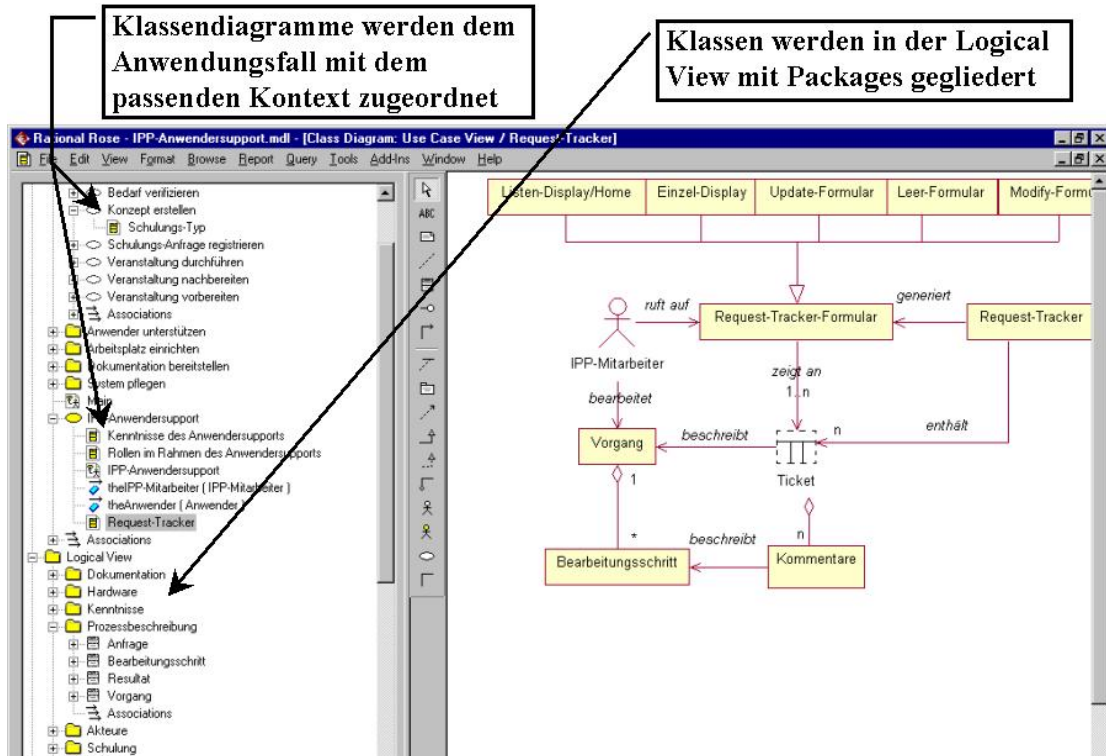


Abbildung 28: Gliederung von Klassen und Klassendiagrammen in Rational Rose

Konventionsvorschläge

- xvi. *Alle Objekte der Prozesse werden durch Klassen repräsentiert und in der Logical View angelegt.*
- xvii. *Klassen werden mit Hilfe von Packages in der Logical View nach formalen Gemeinsamkeiten unabhängig von den Prozessen grob gegliedert.*
- xviii. *Klassen, die innerhalb der Packages ein inhaltliches Cluster bilden, erhalten einen Stereotypen, der das Cluster anschaulich und kurz bezeichnet.*
- xix. *Die Beziehungen der Objekte untereinander sowie zwischen Objekten und Akteuren werden auf Klassendiagrammen dargestellt.*
- xx. *Klassendiagramme werden dem Anwendungsfall zugeordnet, der den engsten noch passenden Kontext für die abgebildete Klassenstruktur darstellt. Im Zweifelsfall werden sie der nächst höheren Ebene zugeordnet.*

4.4.2 Modellierung von Systemoberflächen

Betriebliche Informationssysteme werden idealerweise so konzipiert, dass sie modellübergreifend in mehreren Prozessen zum Einsatz kommen. Für die Modellierung der Systemoberflächen dieser Systeme sind wiederum die Klassendiagramme geeignet. Aus den Klassen abgeleitete Objekte können in Aktivitäts- und Sequenzdiagrammen eingesetzt werden, um die konkrete Verwendung des Systems innerhalb der Prozesse zu demonstrieren.

Die Methoden der Klassen können dazu dienen, die Funktionen zu definieren, die im Rahmen der Prozesse in den Dialogfenstern des Informationssystems angeboten werden oder werden sollen.

Im Falle von IPP ist der Request-Tracker das System, mit dem die Vorgänge sowohl im Prozess „Anwender unterstützen“ als auch im Prozess „Arbeitsplatz einrichten“ verwaltet werden. Die Klassen des *Request-Trackers* müssen also in einem Modell, das beide Prozesse beinhaltet, nur einmal angelegt werden. Die angebotenen Funktionen des *Request-Trackers* können entsprechend als Methode definiert und mehrfach verwendet werden.

Abbildung 29 zeigt die Klasse „Request-Tracker“ mit den Klassen der zugehörigen Formulare incl. der definierten Methoden:

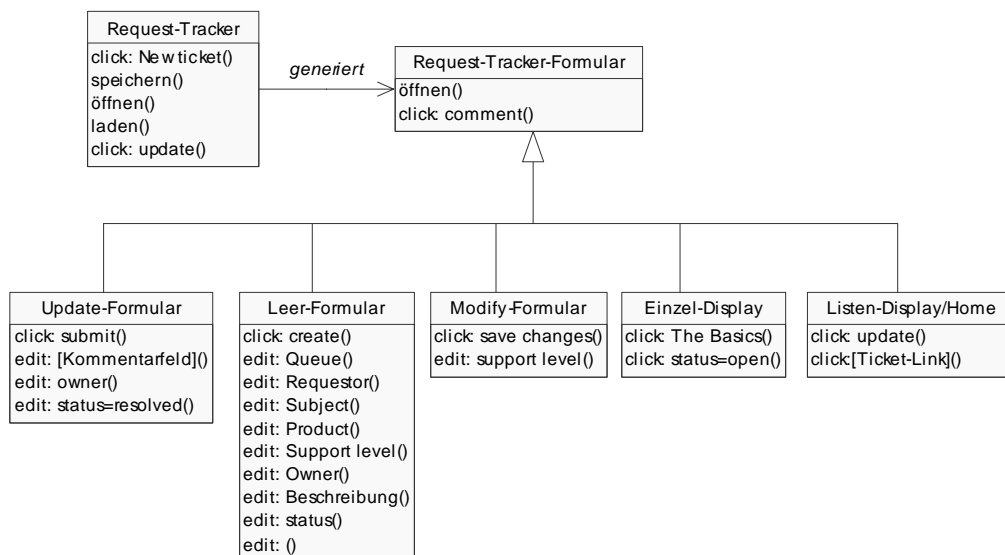


Abbildung 29: Klassen und Methoden bei der Darstellung von Systemoberflächen

Konventionsvorschläge:

- xxi. *Die Objektstruktur von Informationssystemen mit prozessübergreifender Bedeutung wird in einem Klassendiagramm dargestellt, das dem Anwendungsfall der Ebene 0 zugeordnet ist.*
- xxii. *Jedes verwendete Dialogfenster wird durch eine eigene Klasse repräsentiert.*
- xxiii. *Die Anwenderfunktionen in den Dialogfenstern werden durch Methoden der jeweiligen Klassen repräsentiert (s Konventionen xxxiv-xxxvi).*

4.5 Prozesssicht: Geschäftsprozessebene

Sie zeigt die Folge der Aktivitäten eines Prozesses von der Auslösung bis zur Beendigung einschließlich der wichtigsten Verzweigungen. UML-Aktivitätsdiagramme bieten mit ihren Aktivitäten, Übergängen, Verzweigungen, Synchronisationen und Schwimmbahnen die erforderlichen Symbole für die Darstellung der Abläufe.

Ein Ablauf kann als eine bestimmte Verknüpfung der Teilaufgaben einer Aufgabe angesehen werden. Durch die Zuordnung von Aufgaben zu Aktivitäten lässt sich eine Verbindung zwischen Funktions- und Prozesssicht herstellen. Es ist im Sinne einer effizienten Modellierung und Dokumentation, eine vollständige inhaltliche Kongruenz und Benennungsgleichheit zwischen Aufgaben und Aktivitäten desselben Prozesses festzulegen (s. Abbildung 30).

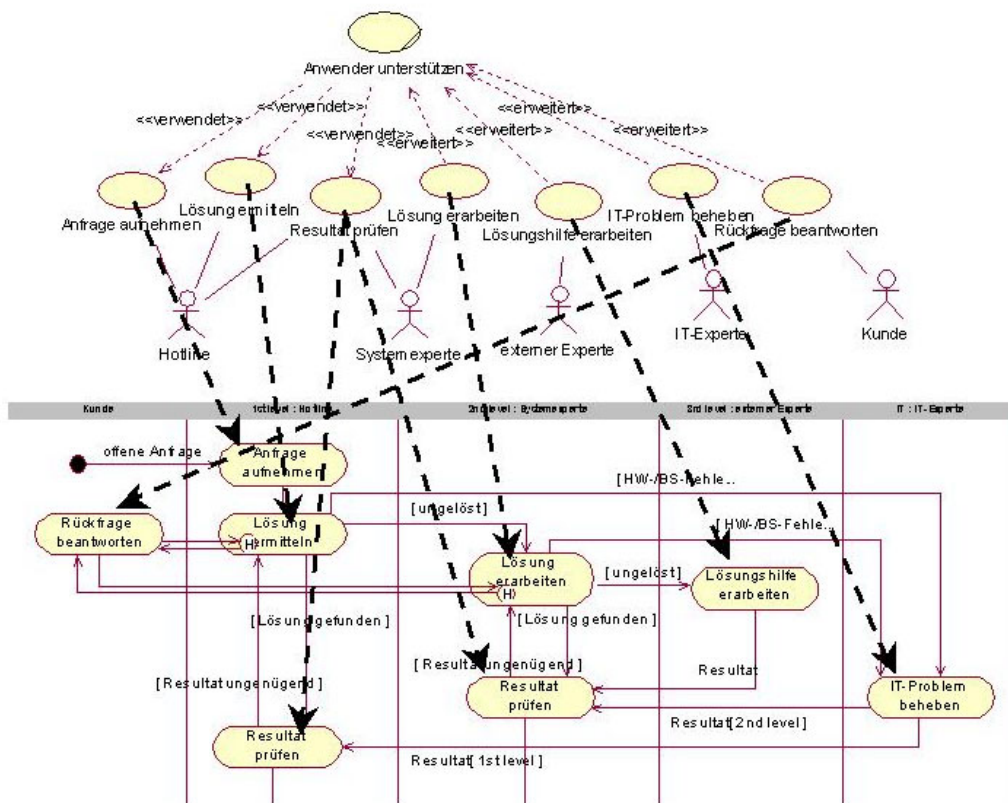


Abbildung 30: Kongruenz von Aufgaben und Aktivitäten

Dadurch kann die Konsistenz und Verständlichkeit des Modells von vornherein gesichert werden, da die gleichen Sachverhalte in den unterschiedlichen Sichten leichter wiederzuerkennen sind.

Konventionsvorschläge:

- xxiv. *Die Ablaufstruktur einer Aufgabe mit definierten Teilaufgaben wird durch ein Aktivitätsdiagramm dargestellt.*
- xxv. *Die Aktivitäten entsprechen jeweils in Benennung und Inhalt genau den Teilaufgaben der Aufgabenstruktur der zu beschreibenden Aufgabe.*
- xxvi. *Akteure in Aktivitätsdiagrammen werden durch Schwimmbahnen repräsentiert, die aus „Actor“- Klassen abgeleitet werden.*
- xxvii. *Objekte in Aktivitätsdiagrammen werden aus Klassen abgeleitet.*

4.6 Prozesssicht: Workflow-Ebene

Die Workflow-Ebene beinhaltet die detaillierte Darstellung der Ausführungsschritte einzelner oder mehrerer Aufgaben in einem konkreten Szenario. Sie umfasst insbesondere die Kommunikation der beteiligten Objekte und Akteure.

4.6.1 Darstellung von Handlungsanweisungen

Ein Geschäftsprozessmodell für die Zielgruppe der Mitarbeiter soll einen praktischen Nutzen im Arbeitsalltag anbieten. Daher sollte es auch detaillierte Handlungsanweisungen für die Bewältigung von Vorgängen umfassen. Insbesondere bei der Einarbeitung neuer Mitarbeiter und bei der Einführung neuer Informationssysteme können diese als Referenz verwendet werden. Für die konkrete Beschreibung eines Vorgangs im Rahmen der UML bieten sich am ehesten Sequenzdiagramme an. Sie stellen die einzelnen Interaktionen zwischen Objekten und Akteuren auf einer Zeitschiene dar (s. nachfolgende Abbildungen). Somit können einerseits die Bedienung von Software-Systemen als auch die Kommunikation zwischen Akteuren berücksichtigt werden.

Bei der konsistenten Erstellung der Sequenzdiagramme müssen der Anspruch einer konsistenten Modellierung und der Anspruch, eine verständliche Handlungsanweisung zu erstellen, in Einklang gebracht werden, was im Folgenden erläutert wird.

Konventionsvorschläge:

- xxviii. *Die Szenarien einer Ablaufstruktur und die Kommunikation beteiligter Objekte und Akteure bei der Ausführung einer Aufgabe werden durch Sequenzdiagramme dargestellt.*
- xxix. *Akteure und Objekte auf Sequenzdiagrammen werden aus Klassen abgeleitet.*

4.6.2 Formale Definition der Nachrichten

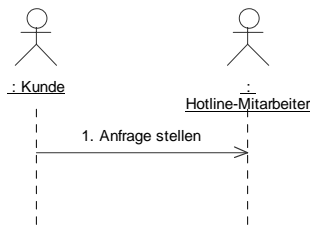


Abbildung 31: Nachricht als Aktion des Senders

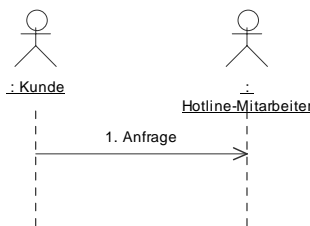


Abbildung 32: Nachricht mit übermitteltem Objekt

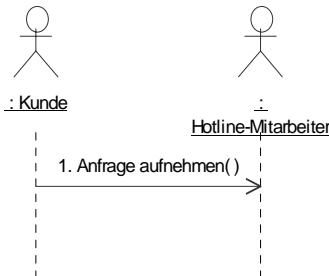


Abbildung 33 :Nachricht mit Benennung aus Empfängersicht

Jeder Bearbeitungsschritt wird in einem Sequenzdiagramm durch eine Nachricht abgebildet. Nachrichten besitzen eine Richtung vom Sendeobjekt hin zum Empfängerobjekt sowie eine Bezeichnung, die den Nachrichten-Inhalt benennt.

Bei der Benennung des Nachrichten-Inhalts muss unterschieden werden zwischen Aktionen des Senders, Methoden des Empfängers und etwaigen Objekten, die übermittelt werden. Die Notation der UML unterscheidet diese Typen nur unzureichend, so dass ohne weitere Regeln widersprüchliche oder zumindest verwirrende Szenarien erstellt werden könnten. Dies ist anhand der Abbildungen 31-33 zu erkennen, die alle denselben Sachverhalt abbilden sollen, und zwar die Anfrage eines Kunden bei einer Hotline.

Abbildung 31 stellt die Aktion des Kunden (Anfrage stellen) in den Vordergrund, während die Nachricht in Abbildung 32 lediglich das übermittelte Objekt (=Anfrage) benennt. Beide unterscheiden sich nicht in der Syntax

Abbildung 33 hingegen benennt die Nachricht aus der Sicht des Empfängers. Dazu wird beim Empfänger die Methode „Anfrage aufnehmen“ ausgelöst, was in der Syntax durch die runden Klammern ausgedrückt wird.

Der darzustellende Bearbeitungsschritt ist somit nicht alleine als Nachricht auf dem Sequenzdiagramm vorhanden, sondern ist auch als Methode der Klasse des Empfängers definiert.

Dies hat den Vorteil, dass die Methode einerseits wiederverwendet werden kann und andererseits das Anforderungsprofil der Klasse transparenter wird.

Zudem kann eine Methode durch Argumente erweitert werden, die beim Aufruf übermittelt werden. Dies ist insbesondere bei der Erfassung einfacher Daten sinnvoll,

um eine Überfrachtung mit zu vielen einzelnen Nachrichten zu vermeiden. Die beiden folgenden Abbildungen verdeutlichen die Alternativen:

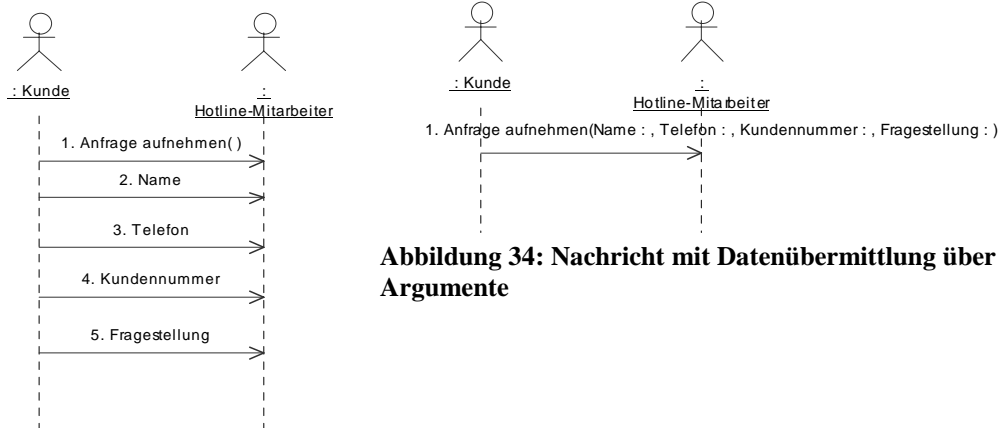


Abbildung 34: Nachricht mit Datenübermittlung über Argumente

Abbildung 35: Szenario mit Nachrichten zur Datenübermittlung

Konventionsvorschläge:

- xxx. Eine Nachricht eines Szenarios ruft eine Methode des Nachrichtempfängers (Objekt oder Akteur) auf, die in der jeweiligen Klasse definiert ist.
- xxxi. Daten, die in einer Nachricht übermittelt werden, werden durch Argumente der jeweiligen Methode dargestellt.

4.6.3 Inhalt und Gliederung der Nachrichten

Durch den Aufruf einer Methode bezieht sich eine Nachricht auf ihren Empfänger, nicht jedoch auf ihren Sender. Daher ist ein Tätigkeitsprofil des Senders auf Basis seiner Methoden zunächst nicht vollständig, da es nicht seine auszusendenden Nachrichten erfasst. Um dies auszugleichen, kann die Aktivität, innerhalb derer die Nachrichten gesandt werden, als Methode des Senders definiert und aufgerufen werden

Abbildung 36 zeigt anhand eines Beispiels aus dem IPP-Prozessmodell, wie die Akteurin „Schulungs-Koordinatorin“ selbst eine eigene Methode aufruft, die der Aktivität „Bedarf verifizieren“ entspricht. Die nachfolgend aufgerufenen Methoden erscheinen als Teilschritte (Aktionen) dieser Aktivität.

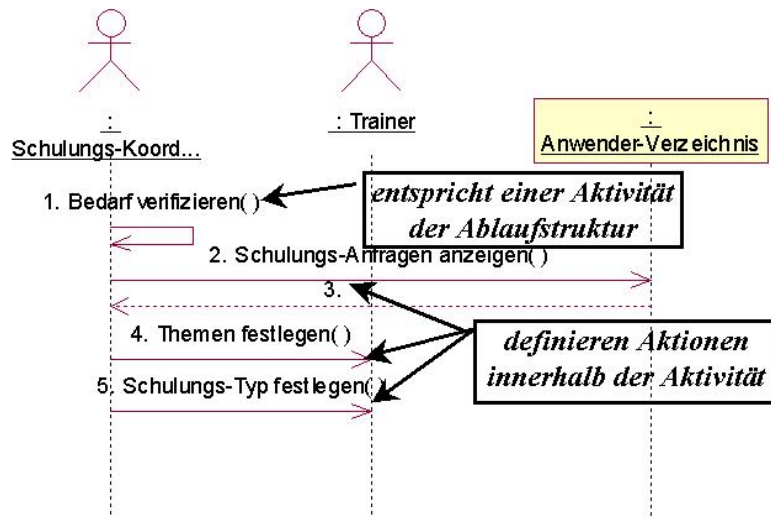


Abbildung 36: Szenario mit Nachrichten verschiedener Detailebenen

Somit wird deutlich, dass Szenarien verschiedene Detaillierungsgrade eines Prozesses umfassen können. Entsprechend werden Methoden definiert, die Tätigkeiten auf verschiedenen Detailebenen widerspiegeln. Es ist zudem im Sinne einer konsistenten und transparenten Darstellung des Zusammenhangs zwischen Workflow-Ebene und Geschäftsprozessebene, dass die Gliederung und Benennung der Nachrichten in den Szenarien die Definition der Aktivitäten in der jeweiligen Ablaufstruktur widerspiegelt. Außerdem sollten für eine transparente Darstellung der Objekte und Akteure die zugehörigen Methoden entsprechend ihrer Detailebene (Aktivität oder Aktion) unterschieden werden. Dazu bietet sich die Verwendung von Stereotypen für die Methoden an.

Konventionsvorschläge:

- xxxii. Eine durch eine Nachricht aufgerufene Methode entspricht
 - entweder nach Inhalt und Benennung einer Aktivität einer Ablaufstruktur („Aktivitäts-Methode“);
 - oder sie definiert eine Aktion innerhalb einer Aktivität („Aktions-Methode“).
 Sie erhält analog einen der Stereotypen „Aktivität“ oder „Aktion“.
- xxxiii. Ein Szenario besteht entweder aus
 - einer Folge von Nachrichten mit Aktivitäts-Methode oder
 - einer Nachricht mit Aktivitäts-Methode, der Nachrichten mit zugehörigen Aktions-Methoden folgen oder
 - aus einer Kombination der beiden ersten Varianten

4.6.4 Bedienung von Informationssystemen

Gerade bei Handlungsanweisungen für den Softwareeinsatz stellt sich die Frage, wie detailliert die einzelnen Schritte wiedergegeben werden.

Geht es darum, einen Vorgang insgesamt oder zumindest mehrere Aufgaben gemeinsam zu betrachten, müssen für das grundsätzliche Verständnis unwesentliche Objekte und Schritte ausgeblendet werden, um eine übersichtliche Handlungsanweisung zu erhalten. Die folgende Abbildung zeigt anhand eines Vorgangs bei IPP die prinzipielle Verwendung eines Tickets im *Request-Tracker*, wobei einzelne Formularobjekte des verwendeten Systems ausgeblendet sind.

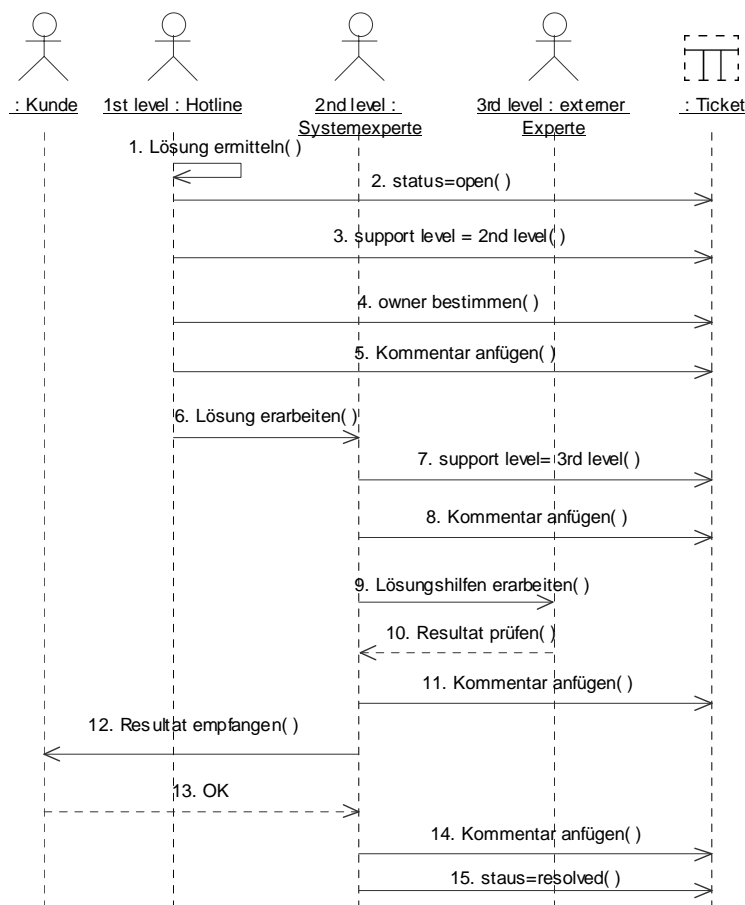


Abbildung 37: Szenario für die Bearbeitung eines Datensatzes unter Ausblendung der Systemoberfläche

Die Darstellung konzentriert sich auf die für den Prozess wesentlichen Operationen, an denen der Mitarbeiter unmittelbar beteiligt ist. Es wird dem Mitarbeiter selbst überlassen, einen eigenen Weg zu finden, in welcher Weise er den hier verborgenen *Request-Tracker* bei der Bearbeitung des Tickets bedient.

Diese Darstellung Abbildung 37 ist für einen ungeübten Erstnutzer des *Request-Trackers* u.U. nicht ausreichend, um einen Weg durch die vielfältige Funktionalität zu finden. Bei der Einarbeitung in die Systemfunktionen ist entscheidend, dass man sich auf der Systemoberfläche orientieren kann, um dort die Schritte der Handlungsanweisungen umsetzen. Diese Anforderung kann dadurch erreicht werden, dass alle elementaren Einzelschritte durch separate Nachricht dargestellt werden und die Benennungen der entsprechenden Methoden konsistent und einheitliche gehalten werden. Dabei sollten die Benennungen der Menüs, Buttons, Hyperlinks und Feldern der jeweiligen Ansicht der Systemoberfläche berücksichtigt werden.

Abbildung 38 zeigt dazu als Beispiel einen Ausschnitt der exakten Verwendung des *Request-Trackers* im Rahmen der Aufgabe „Lösung erarbeiten“.

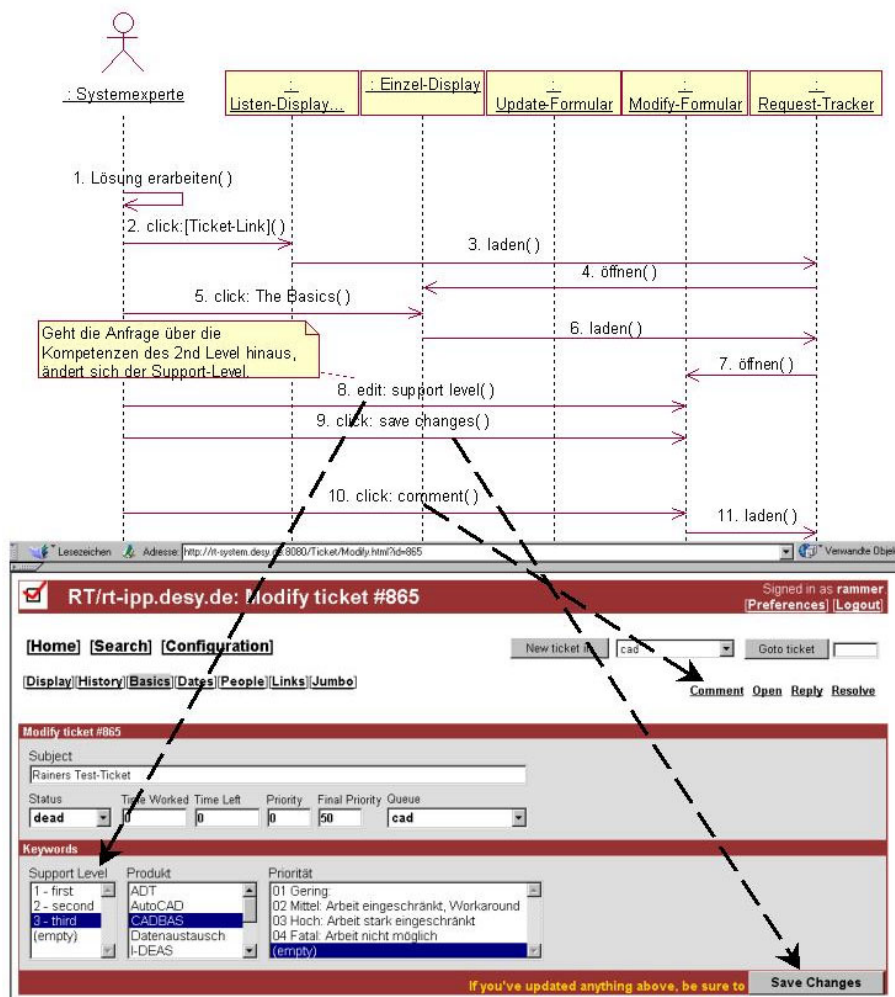


Abbildung 38: Benennung von Anwenderfunktionen in Sequenzdiagrammen

Konventionsvorschläge:

- xxxiv. *Die Aktivierung eines Buttons oder Hyperlinks auf einer Systemoberfläche wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „click: [Button- oder Hyperlink-Name]()“ aufruft.*
- xxxv. *Die Aktivierung einer Menüfunktion wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „click: [menüpfad]/[menüfunktion]()“ aufruft.*
- xxxvi. *Die Änderung eines Eingabe- oder Auswahlfeldes auf einer Systemoberfläche wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „edit: [Feldname]=[Eingabe]()“ aufruft.*

4.7 Weitere Anforderungen der Prozessdokumentation

4.7.1 Beschreibungstexte

Die Definition des Modells erfolgt im Wesentlichen über die Diagramme mit den abgebildeten Elementen und Beziehungen. Diese können jedoch nie vollständig selbsterklärend sein. Es bedarf weiterer Erläuterungen, die einen Zweck, eine Bedeutung oder einen Kontext näher beschreiben. In *Rational Rose* können dafür Texte in das Feld „Documentation“ der Element-Spezifikation eingefügt werden.

Die Pflege dieser Texte ist ein bedeutender paralleler Aufwand, da sie mit der Benennung und Bedeutung der Modellelemente im Einklang stehen und bei Modelländerungen neu angepasst werden müssen. Daher ist im Sinne einer effizienten Modellverwaltung die Verwendung dieser Texte möglichst überschaubar zu gestalten, d.h. dass nur bestimmte Elementtypen eine Beschreibung erhalten. Unentbehrlich ist eine Beschreibung der Anwendungsfälle, da sie die Basisstruktur darstellen. Durch die kongruente Bedeutung von Aufgaben, Aktivitäten und Aktivitäts-Methoden der Methoden sind die Texte zudem gleichermaßen für die Funktions- und Prozesssicht verwendbar. Daneben sollten Klassen, Rollen und Aktions-Methoden immer dann eine Beschreibung erhalten, wenn ihre Bedeutung nicht aus der Benennung und Verwendung bereits klar hervorgeht, denn sie stellen die Verankerung des Modells in den Objekten, Akteuren und Arbeitsschritten der betrieblichen Realität dar.

Eine vollständige Dokumentation erfordert zudem eine Präambel, die den Kontext und die Entstehung des Modells zusammenfassend beschreibt.

Konventionsvorschläge:

- xxxvii. Die Präambel des Modells wird im Dokumentationsfeld der „Use Case View“ niedergelegt.*
- xxxviii. Jede Aufgabe erhält obligatorisch eine Beschreibung im Dokumentationsfeld des jeweiligen Anwendungsfalles.*
- xxxix. Objekte, Akteure und Methoden erhalten eine Beschreibung im Dokumentationsfeld der jeweiligen Klassen und Methoden. Davon kann abgesehen werden, wenn die Bedeutung offensichtlich oder aus dem Kontext leicht zu erschließen ist.*

4.7.2 Steuerung der Print-Dokumentation

Eine weitere wesentliche Anforderung ergibt sich aus der linearen Darstellung eines Modells in einer Print-Dokumentation. Theoretisch kann aus den Abläufen eine Reihenfolge der Aufgaben abgeleitet werden; dies lässt sich jedoch durch die Exportfunktionen von *Rational Rose* nicht formal abbilden. Zudem wird dabei noch nichts über die Darstellungsreihenfolge optionaler oder paralleler Aufgaben sowie der gesamten Prozesse untereinander ausgesagt. Ebenso muss die primäre Zuordnung von Aufgaben zu Prozessen erkennbar sein, um bei einer Mehrfachverwendung entsprechende Querverweise in die Print-Dokumentation einfügen zu können. Die Verwendung des Feldes „rank“ in der Spezifikation der Anwendungsfälle ermöglicht dabei eine flexible Steuerung der Print-Dokumentation. Durch eine entsprechende Kodierung kann die Reihenfolge aller Anwendungsfälle sowohl innerhalb eines Prozesses als auch in Bezug auf das gesamte Modell festgelegt werden.

Konventionsvorschläge:

- xl. Jede Aufgabe erhält eine dreiteilige numerische Kennung nach dem Muster xxyyzz im Feld „Rank“ des jeweiligen Anwendungsfalles. Der erste Teil (xx) kennzeichnet den Geschäftsprozess, dem die Aufgabe primär zugeordnet ist, der zweite Teil (yy) die Ebene in der Hierarchie und der dritte Teil (zz) benennt den Rang der Bedeutung innerhalb einer Ebene eines Geschäftsprozesses. Der Anwendungsfall der Ebene 0 erhält die Kennung „000000“.*

4.7.3 Einbindung von Dokumenten

Eine wesentliche Anforderung an die Dokumentation eines Prozessmodells ist die Verknüpfung zu Dokumenten, die sich außerhalb der Modelldatei befinden, aber nützliche Hilfs- und Arbeitsmittel für die praktische Anwendung sind. *Rational Rose* bietet hierzu die Möglichkeit, Verweise an Elemente anzuhängen. Für eine übersichtliche Verwaltung ist es jedoch sinnvoll, die Zahl der so erweiterten Elemente zu beschränken, z.B. indem man diese Option nur für die Elemente zulässt, die als Referenz bei der praktischen Umsetzung besonders bedeutsam sind.

Konventionsvorschläge:

xli. Dateien und Intranetquellen mit Hilfs- und Arbeitsmitteln werden immer entweder Szenarien, Aufgaben oder Objekten zugeordnet. Entsprechende Verweise werden daher den jeweiligen Sequenzdiagrammen, Anwendungsfällen oder Klassen angehängt.

5 Die Anwendersicht des Publikationssystems

In diesem Kapitel steht die Beschreibung des Aufbaus und der Funktionsweise des Publikationssystems aus Anwendersicht im Vordergrund. Ausgehend von den Nutzungs-Anforderungen und den Kriterien der Gebrauchstauglichkeit werden die Gestaltungsmerkmale der Systemoberfläche vorgestellt. Die praktische Anwendung des Systems wird anhand von realitätsnahen Szenarien beschrieben.

5.1 Anforderungen und Kriterien der Anwendersicht

5.1.1 Verwendung und Nutzen des Systems

Die Anwendersicht ist entscheidend für den Mehrwert des Systems. Über sie erhalten die Zielgruppen des Systems Zugang zu den Modellinhalten. Diese sollen in einer Weise vermittelt werden, die das Verständnis über das Modell fördert.

Im Einzelnen soll das System folgende Nutzungsmöglichkeiten für die verschiedenen Anwendergruppen bieten (s.a. Abschnitt 2.2):

Aus Sicht eines Gruppen-/Abteilungsleiters:

- Überblick über Prozessabläufe;
- Überblick über benötigte / eingesetzte Arbeitsmittel;
- Ansicht von Tätigkeitsprofilen der Akteure um z.B. Stellenbeschreibungen anfertigen zu können;
- Darstellung der Dienstleistungen und Arbeitsweise bei Präsentationen;
- Referenz zur Verfolgung von Informations- / Formular- / Entscheidungswegen;
- Arbeitsgrundlage für Prozessoptimierungen;
- Arbeitsgrundlage für Abstimmungen (z.B. an Schnittstellen) mit anderen Gruppen;
- Zentrale Quelle aller Prozess-Informationen.

Aus Sicht eines Prozessverantwortlichen:

- Beschreibung des Prozessablaufs, z.B für Präsentationen und als Diskussionsmaterial;
- Überblick über die Zuständigkeiten im Prozess;
- Zugriff auf Arbeitsmaterialien für Aktualisierungen.

Aus Sicht eines Mitarbeiters im Prozess:

- Erinnerung an den Prozessablauf;
- Checkliste für alle durchzuführenden Tätigkeiten;

- Zugriff auf benötigte Arbeitsmaterialien;
- Inhaltliche Beschreibung der Aufgaben (im Sinne eines „Referenzhandbuchs“);
- Abfolge und Beschreibung der Arbeitsschritte;
- Informationen über die eigenen Zuständigkeiten und die anderer Prozessbeteiligter.

Aus Sicht eines Kunden:

- Verbale Beschreibungen der Dienstleistungen;
- Hinweise über Zuständigkeiten;
- Hinweise und Hilfsmittel zur Anforderung von Dienstleistungen.

Die genannten Anforderungen werden durch die folgenden drei Funktionen der Anwendersicht (s. Abbildung 39) implementiert.

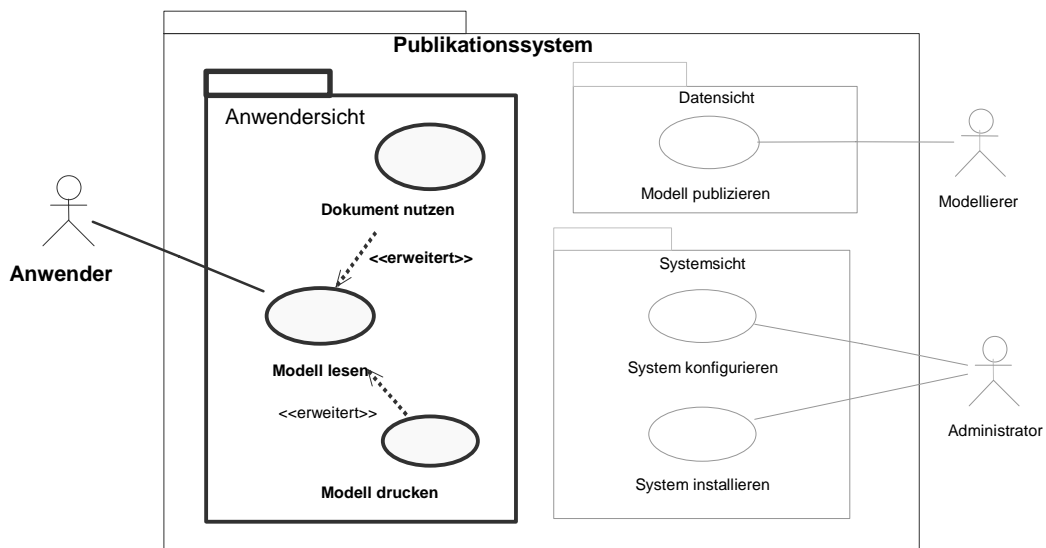


Abbildung 39: Funktionen der Anwendersicht des Publikationssystems

Die Funktion "Modell lesen" ermöglicht es dem Anwender, die Grundzüge und Details eines Modells vollständig zu erfassen, indem er in interaktiver Weise durch das Modell navigiert.

Die Funktion "Dokument nutzen" bietet den Zugang zu Hilfsmitteln und Arbeitsmaterialien, die einzelne Aufgaben, Szenarien und Objekte des publizierten Geschäftsprozess-Modells näher beschreiben bzw. Unterstützung im Arbeitsalltag geben.

Mit der Funktion "Modell drucken" erhält der Anwender die Möglichkeit, eine Print-Fassung des gesamten Modells oder eines relevanten Ausschnitts zu erstellen.

5.1.2 Kriterien der Gebrauchstauglichkeit

Gerade die Vermittlung komplexer Inhalte verlangt, dass bei der Gestaltung einer Benutzeroberfläche eine zufriedenstellende Gebrauchstauglichkeit hergestellt wird. Es soll sichergestellt werden, dass der zu vermittelnde Inhalt und nicht das Design im Vordergrund steht. Desweiteren ergeben sich aus ihnen Anforderungen für eine nutzenorientierte und anwenderfreundliche Funktionalität.

Ausgangspunkt für die Kriterien ist die Norm DIN EN ISO 9241-11, die Gebrauchstauglichkeit wie folgt definiert:

„Das Ausmass, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und mit Zufriedenheit zu erreichen.“ (NORM 1995)

Der Nutzungskontext des Publikationssystems ergibt sich aus dem darzustellenden Inhalt (Geschäftsprozessmodelle), den Zielen (Transparenz, Unterstützung im Arbeitsalltag etc.), der Zielgruppe (Mitarbeiter und Kunden) sowie den technischen Gegebenheiten (Intranet).

Effektiv ist die Nutzung, wenn die angestrebten Ziele der Systemangebote durch den Anwender genau und vollständig erreicht werden können.

Effizient ist die Nutzung, wenn der Aufwand der Anwender gegenüber der Genauigkeit und Vollständigkeit der Zielerreichung möglichst gering ist.

Zufriedenstellend ist die Nutzung, wenn der Anwender keine Beeinträchtigungen bei der Nutzung erlebt und eine positive Einstellung gegenüber dem System entwickelt.

(vgl. Zimmermann 2002)

Diese generellen Kriterien erhalten eine Konkretisierung durch die Grundsätze der Dialoggestaltung für Softwaresysteme, die in der Norm DIN EN ISO 9241-10 niedergelegt sind:

- Aufgabenangemessenheit
- Selbstbeschreibungsfähigkeit
- Steuerbarkeit
- Erwartungskonformität
- Fehlertoleranz
- Individualisierbarkeit
- Lernförderlichkeit

(Norm 1996)

5.2 Aufbau der Anwendersicht

5.2.1 Gestaltung der Systemoberfläche

Im Folgenden wird erläutert, wie die Kriterien der Gebrauchstauglichkeit in der Anwendersicht des Publikationssystems umgesetzt worden sind.

Steuerbarkeit: Die Steuerung durch den Anwender erfolgt hauptsächlich anhand der vorhandenen Navigationsbereiche. Dabei wird zwischen drei verschiedenen Navigationsbereichen differenziert (s. Abbildung 40):

Die **Systemnavigation** besteht modellunabhängig, ermöglicht den Wechsel des Modells, bietet Hilfen an und zeigt den Ausgang aus dem System.

Die **Modellnavigation** ermöglicht die Navigation zwischen den verschiedenen Sichten innerhalb der Basisstruktur des Modells (s. Abschnitt 4.1) und verwendet für die Verweise Benennungen, die direkt aus dem Modell selbst stammen.

Die **Aufgabennavigation** ermöglicht die Navigation ausgehend von und innerhalb der aktuellen Position in der Basisstruktur des Modells.

Zusätzlich kann über einen Verweis in der Kontextbeschreibung der Anwendungsfälle und Aktivitäten zu einer anderen Aufgabe gewechselt werden.

Als weiteres Navigationsmittel kann eine Sitemap dienen, die den gesamten Baum der Modellhierarchie in Hyperlink-Form abbildet .

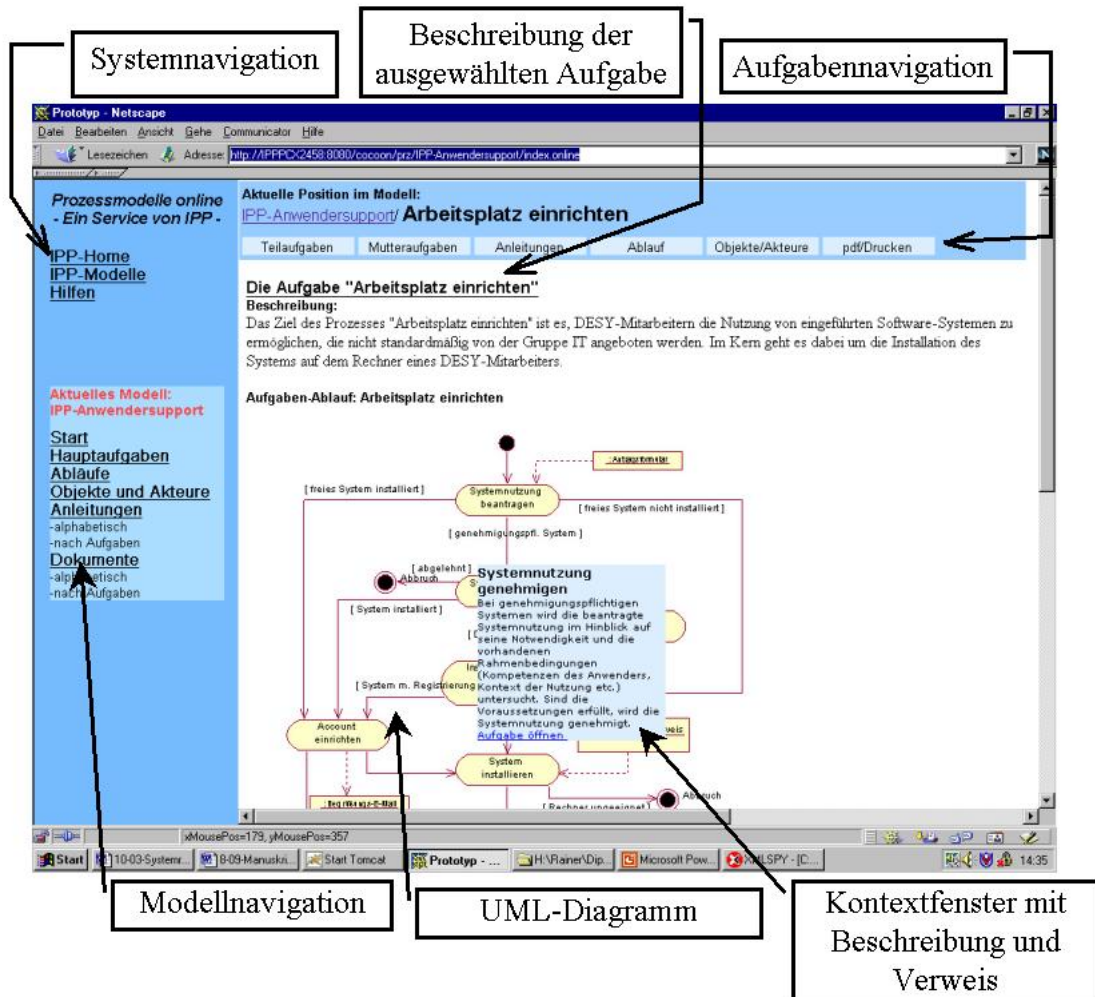


Abbildung 40: Navigation und Gliederungsbereiche der Systemoberfläche

Aufgabenangemessenheit : Die Aufgaben der Mitarbeiter werden über die in einem Modell enthaltenen Arbeitsanleitungen und Dokumente unterstützt. Diese sind sowohl über die Aufgabennavigation bzw. entsprechende Diagrammelemente als auch direkt über Gesamtlisten erreichbar. Die aktuelle Aufgabe kann insgesamt oder nur mit einzelnen Diagrammen ausgedruckt werden.

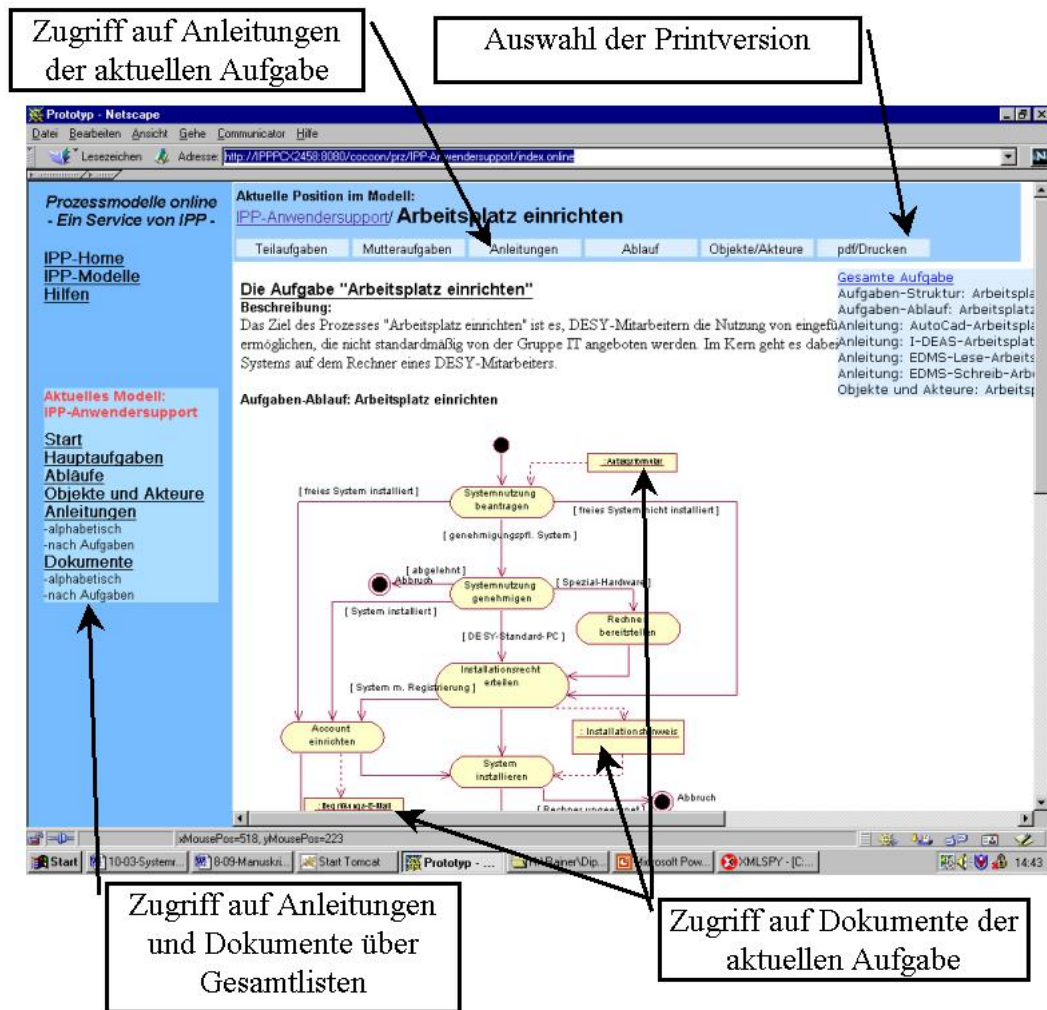


Abbildung 41: Zugriffe der Systemoberfläche auf Anleitungen, Dokumente und Printversionen

Selbstbeschreibungsfähigkeit : Das aktuell ausgewählte Modell wird in hervorgehobener Schrift genannt. Die Orientierung innerhalb des Modells wird dadurch unterstützt, dass der Pfad der aktuell ausgewählten Aufgabe innerhalb der Modellhierarchie im Bereich der Aufgabennavigation angezeigt wird (s. Abbildung 43).

Darüberhinaus wird das Erstellungs- und das letzte Aktualisierungsdatum erscheinen.

Erwartungskonformität: In verschiedenen Modellen gelangen Verweise mit den gleichen Benennungen zu Modellausschnitten bzw. Diagrammen mit analoger Bedeutung.

Die Benennungen der Modellsichten und –diagramme wurde an den alltäglichen Sprachgebrauch angepasst. Die folgende Tabelle gibt dazu einen Überblick:

Benennungen der Fachsprache	Benennungen im Publikationssystem
Funktionssicht (Anwendungsfalldiagramme)	(Haupt-)aufgaben; Teilaufgaben
Organisationssicht (Klassendiagramme)	Akteure
Prozesssicht auf Geschäftsprozessebene (Aktivitätsdiagramme)	Abläufe
Prozesssicht auf Workflowebene , Szenarien (Sequenzdiagramme)	Anleitungen
Objektsicht (Klassendiagramme)	Objekte, Arbeitsmaterialien, Hilfsmittel, Dokumente

Abbildung 42: Angepasste Benennungen im Publikationssystem

Lernförderlichkeit: Das System stellt geringe Lernanforderungen an den Anwender, da es eine intuitive Nutzung ermöglicht, die durch einfache Erläuterungen eines Hilfetextes und durch ein Glossar unterstützt wird.

Wesentlich für die intuitive Handhabung ist die verweissensitive Gestaltung der Diagrammelemente. Durch Anklicken wird die Element-Beschreibung aufgerufen und etwaige Verweise auf Dokumente werden aufgelistet. Bei Aufgaben- und Aktivitätselementen wird zudem die Möglichkeit geboten, die Ansicht auf die gewählte Aufgabe zu wechseln.

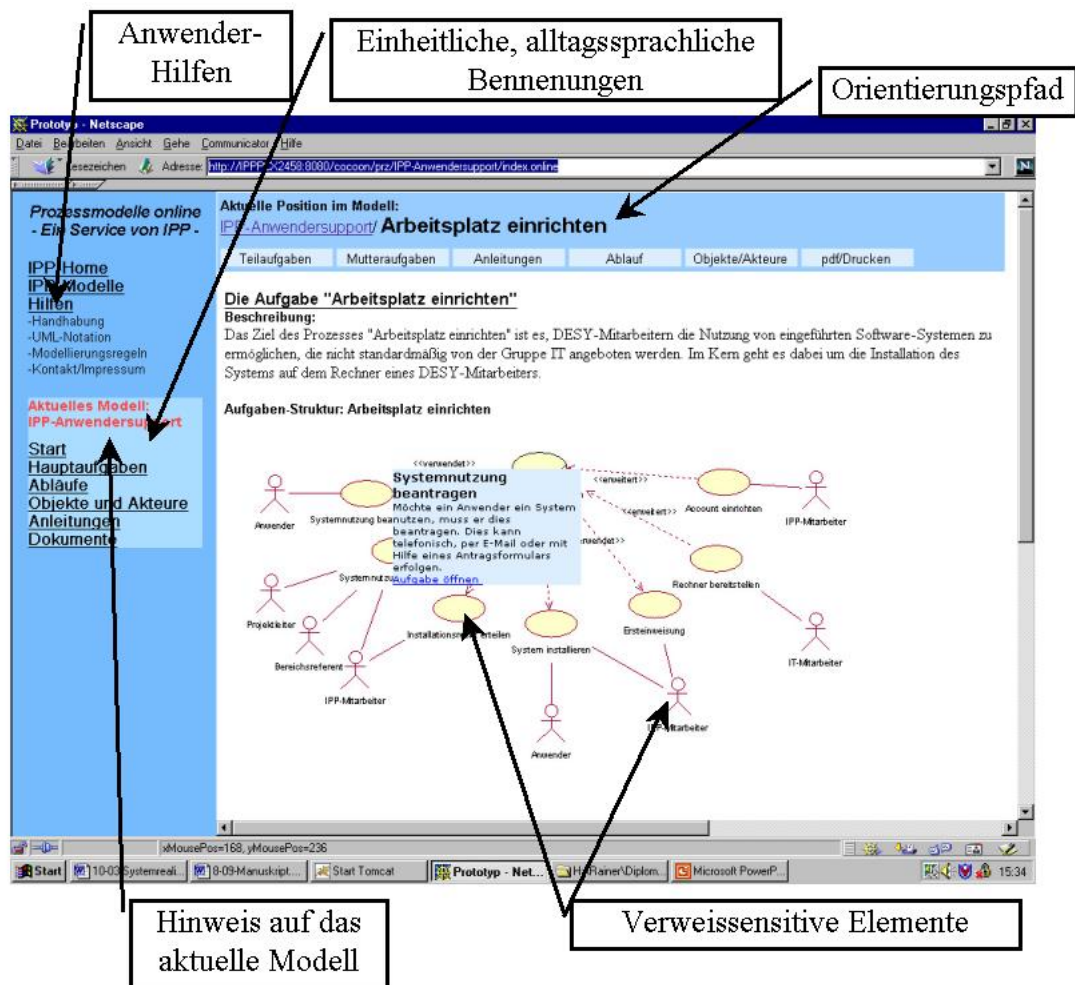


Abbildung 43: Unterstützung einer intuitiven Orientierung und Nutzung der Systemoberfläche

Fehlertoleranz: Dieses Kriterium ist von begrenzter Relevanz, da seitens des Systems keine Eingaben vorgesehen sind, die über die Benutzung von Funktionen und Verweisen hinausgehen. Die Wiederherstellung vorheriger Zustände nach irrtümlichen Aktionen ist durch die transparente Gestaltung der Navigation bzw. durch die Browserfunktionalität ausreichend sichergestellt.

Individualisierbarkeit: Da der Zweck des Systems nicht im Dauereinsatz bei der Bewältigung einzelner Arbeitsaufgaben besteht, sondern in der allgemeinen Unterstützung der betrieblichen Vorgänge im Sinne einer Referenzdokumentation, ist auch dieser Grundsatz nur begrenzt relevant. Bei einer großen Anzahl von publizierten Modelle könnte die Handhabung dadurch komfortabler werden, dass Benutzerkonten mit persönlichen Modell- und Diagrammlisten eingeführt werden.

Zudem wären Individualisierungen hinsichtlich der Default-Ansicht sowie der Farb- und Schriftgestaltung denkbar.

5.2.2 Gestaltung der Printfassung

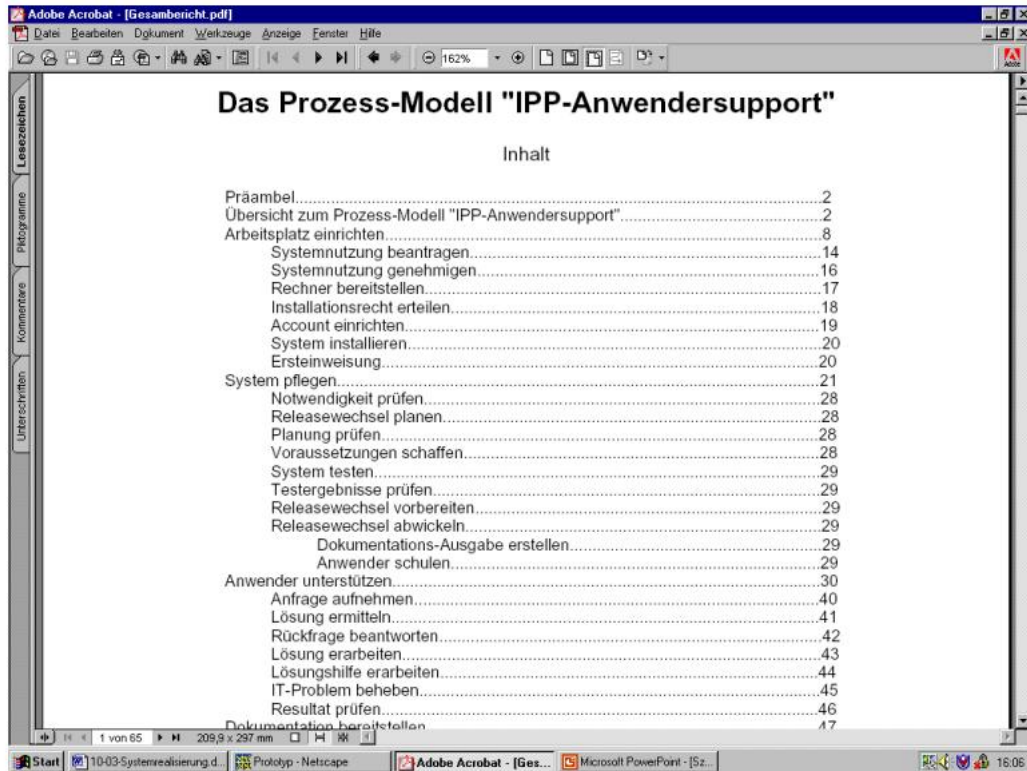
Der Anwender soll die Möglichkeit haben, zwischen verschiedenen Darstellungsformen zu wählen, um Informationen über einen Geschäftsprozess gemäß einem bestimmten Zweck oder persönlichen Vorlieben angemessen auswählen und aufnehmen zu können. Daher wird mit dem Publikationssystem auch die Option geboten, das Modell zusätzlich auszudrucken.

Die gesamte Dokumentation eines Modells einschließlich aller Diagramme und Element-Beschreibungen kann von einigen dutzend bis zu mehreren hundert Druck-Seiten umfassen. Meist wird jedoch nur der Ausdruck eines Geschäftsprozesses, einer Aufgabe oder sogar nur eines einzelnen Diagramms benötigt werden. Die Auswahl wird immer von dem konkreten Nutzungsbedarf in der aktuellen Arbeitssituation eines Anwenders abhängen. Dieser kann z.B. aus der Einarbeitung in eine neue Stelle, aus der Bedienung eines Informationssystems oder aus der kritischen Analyse der eigenen Vorgangsbearbeitung bestehen.

Im Sinne der Konsistenz zwischen Modellstruktur und ihrer Abbildung im Publikationssystem orientieren sich die Druckoptionen im System ebenfalls an der Basisstruktur des Modells. Der Anwender erhält mit der „Drucken“-Schaltfläche in der Aufgabennavigation die Möglichkeit, die aktuelle Aufgabe auszudrucken. Dabei hat er die Wahl zwischen der gesamten Aufgabe und den einzelnen Diagrammen der Aufgabe.

Der innere lineare Aufbau einer Printfassung erfolgt nach einem festgelegten Schema, das die Bedeutung der verschiedenen Sichten widerspiegeln soll. Dabei muss auch der hierarchische Bezug von Aufgaben zu Teilaufgaben beachtet werden, d.h. der Ausdruck einer gesamten Aufgabe beinhaltet den Ausdruck der zugeordneten Teilaufgaben aller niedriger liegenden Ebenen der Modellhierarchie. Da somit in einem Ausdruck eine mehrfache iterative Verschachtelung möglich ist, erscheinen aus Gründen der leichteren Orientierung zunächst alle Diagramme und Elemente einer Aufgabe vor der Darstellung der Teilaufgaben.

Die Reihenfolge der Aufgaben und Teilaufgaben wird dabei durch den vom Modellierer festgesetzten Rang der Anwendungsfälle gesteuert. Diese hierarchische Abfolge bildet auch die Basis des Inhaltsverzeichnisses.



Inhalt	
Präambel.....	2
Übersicht zum Prozess-Modell "IPP-Anwendersupport".....	2
Arbeitsplatz einrichten.....	8
Systemnutzung beantragen.....	14
Systemnutzung genehmigen.....	16
Rechner bereitstellen.....	17
Installationsrecht erteilen.....	18
Account einrichten.....	19
System installieren.....	20
Ersteinweisung.....	20
System pflegen.....	21
Notwendigkeit prüfen.....	28
Releasewechsel planen.....	28
Planung prüfen.....	28
Voraussetzungen schaffen.....	28
System testen.....	29
Testergebnisse prüfen.....	29
Releasewechsel vorbereiten.....	29
Releasewechsel abwickeln.....	29
Dokumentations-Ausgabe erstellen.....	29
Anwender schulen.....	29
Anwender unterstützen.....	30
Anfrage aufnehmen.....	40
Lösung ermitteln.....	41
Rückfrage beantworten.....	42
Lösung erarbeiten.....	43
Lösungshilfe erarbeiten.....	44
IT-Problem beheben.....	45
Resultat prüfen.....	46
Dokumentation bereitstellen.....	47

Abbildung 44: Inhaltsgliederung einer Printversion

Innerhalb einer Aufgabe erscheint zunächst die Aufgabenstruktur als primäre Sicht gemäß der vorgeschlagenen Modellkonventionen (s. Abschnitt 4.1).

Es folgen daraufhin die Ablaufstruktur, Anleitungen sowie Objekte und Akteure in der genannten Reihenfolge.

Durch die Mehrfachverwendung von Aufgaben, vor allem aber von Klassen und Methoden besteht leicht die Gefahr von redundanten Beschreibungen an verschiedenen Stellen eines umfassenden Modellausdrucks. Dies kann im Einzelfall sinnvoll sein, bläht den gesamten Umfang jedoch auf, so dass dem Anwender die Orientierung erschwert wird. Die Wiedergabe von Element- und Aufgabenbeschreibungen wurde daher auf Aktivitäts- und Klassendiagramme beschränkt.

5.3 Anwenderszenarien

Das folgende Diagramm beschreibt den generalisierten Nutzungsablauf aus Sicht des Anwenders. Dabei wird von einer intuitiven Navigation vom Kontext (Gesamtmodell) zum Detail (Modellelement) ausgegangen. Die Nutzung von Dokumenten und der Ausdruck des Modells sind dabei optionale Aktivitäten.

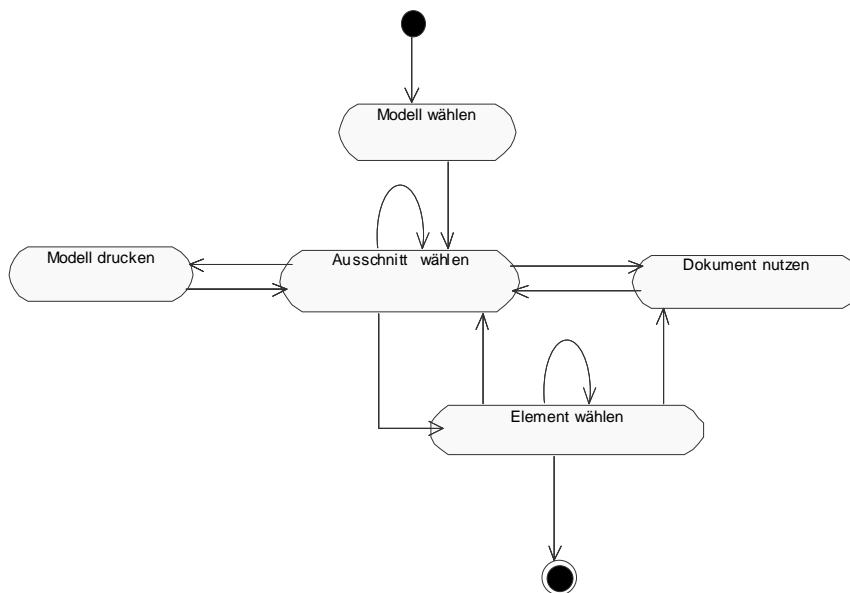


Abbildung 45: Ablaufstruktur der Funktion „Modell lesen“

Die folgenden Szenarien innerhalb dieses Ablaufs demonstrieren den praktischen Nutzen des Publikationssystems.

5.3.1 Szenario 1: Bewältigung eines Vorgangs im Prozess „Anwender schulen“

Ausgangssituation:

Ingrid R. sei eine neue Mitarbeiterin bei IPP. Sie übernimmt die Koordination der Anwenderschulungen. Bei ihr gehen Anfragen nach Schulungen ein, die sie sammelt und weiterverfolgt. Zudem ist sie an der Vorbereitung, Durchführung und Nachbereitung der Schulungen beteiligt. Während ihrer Einarbeitungszeit verwendet sie das Publikationssystem, um die einzelnen Arbeitsschritte und -materialien kennzulernen.

Umsetzung:

Sie geht auf die Startseite des Modells und erhält eine Übersicht über die Hauptaufgaben des Anwendersupports. Sie wählt die Hauptaufgabe „Anwender schulen“ und klickt auf den entsprechenden Anwendungsfall (1). Im damit geöffneten Kontextfenster liest sie die Beschreibung der Hauptaufgabe und sie klickt auf „Aufgabe öffnen“, um weitere Details zu erfahren (2).

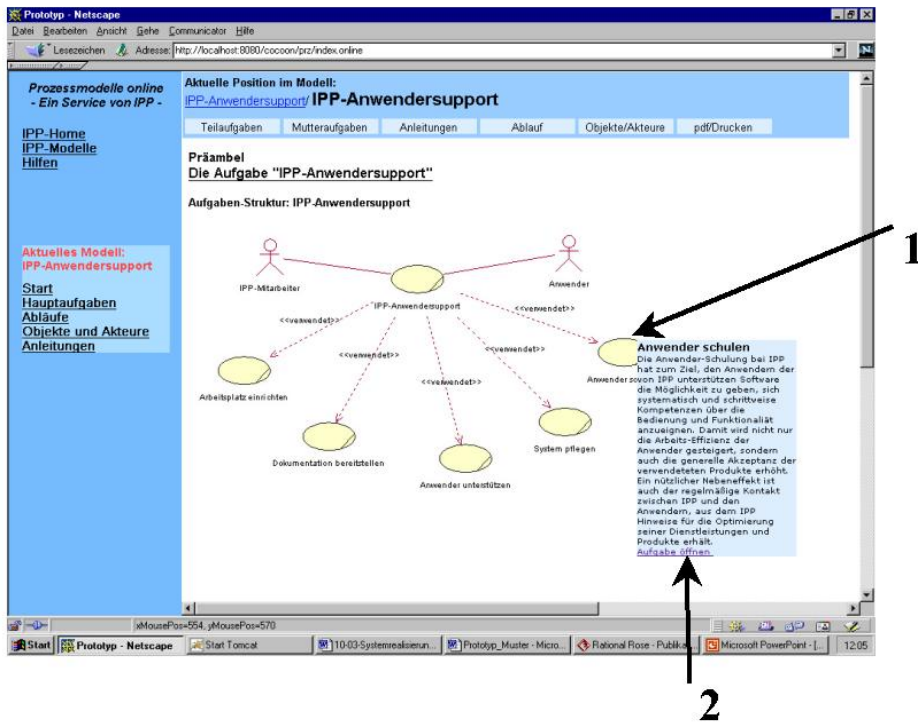


Abbildung 46: Szenario 1, Schritte 1 und 2

Es erscheint die Aufgabenstruktur von „Anwender schulen“ und sie klickt auf "Veranstaltung nachbereiten" (3), um Einzelheiten über die einzelnen Schritte dieser Aufgabe zu erhalten. Das Kontextfenster liefert eine Kurzbeschreibung. Wiederum klickt sie auf "Aufgabe öffnen" (4), um eine Detailansicht zu erhalten.

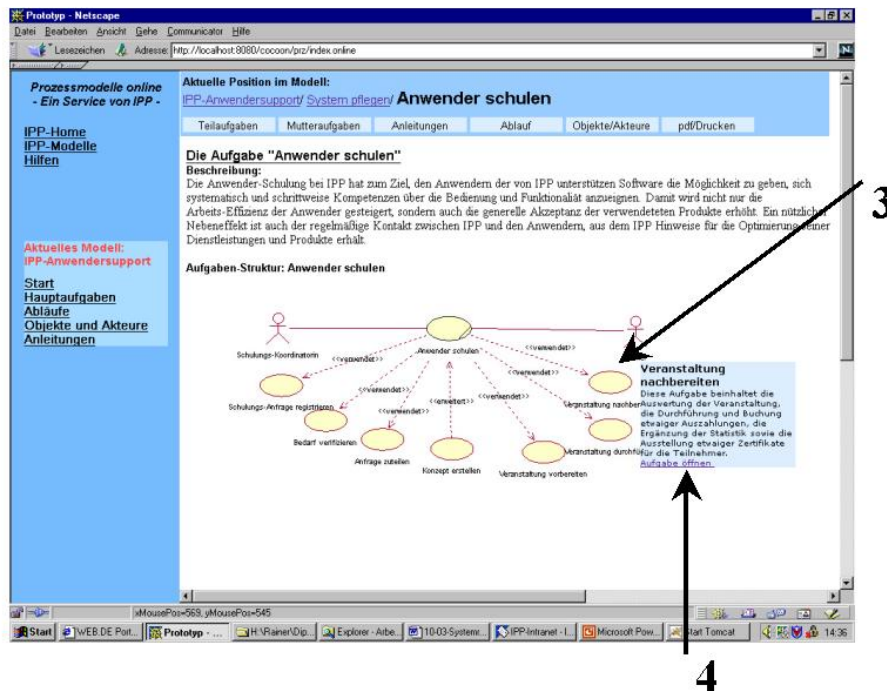


Abbildung 47: Szenario 1, Schritte 3 und 4

Nun erscheinen die Anleitungen für die Nachbereitung der unterschiedlichen Veranstaltungen (hier: IDEAS-Unterricht). Dort kann sie die notwendigen Tätigkeiten ablesen (5). Durch Anklicken der dargestellten Symbole für Arbeitsmaterialien gelangt sie zu deren Kurzbeschreibungen (6). Diese können mit weiteren Links direkt mit den zugehörigen Dateien (z.B. ein Teilnehmerverzeichnis) verknüpft werden. Die Druckfunktion des Systems (7) ermöglicht ihr, einen Ausdruck der Anleitung zu erstellen (8), um sie an ihre Pinnwand zu heften.

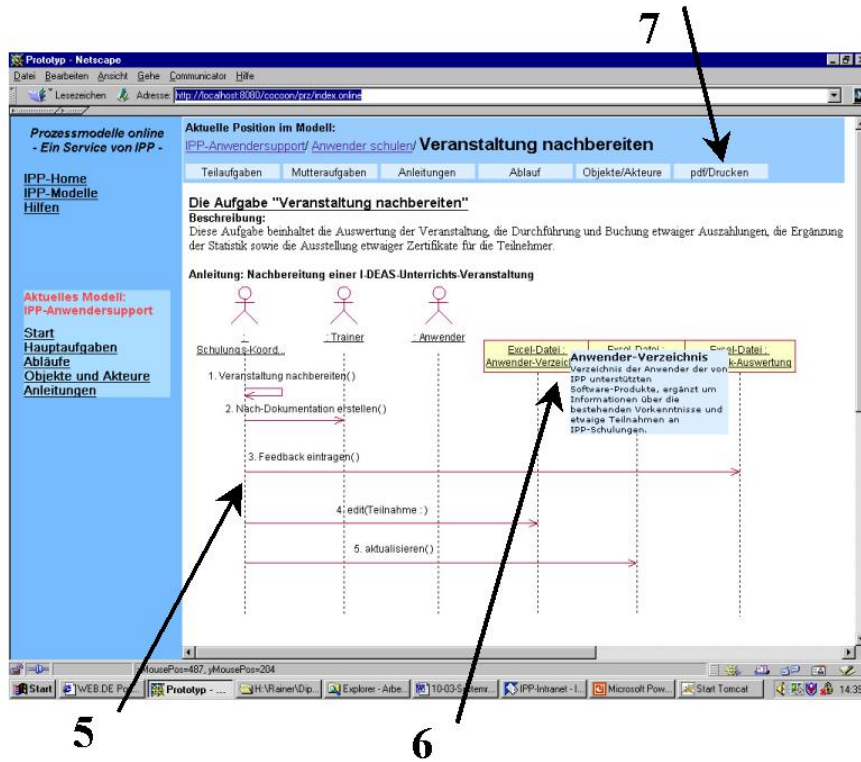


Abbildung 48: Szenario 1, Schritte 5-7

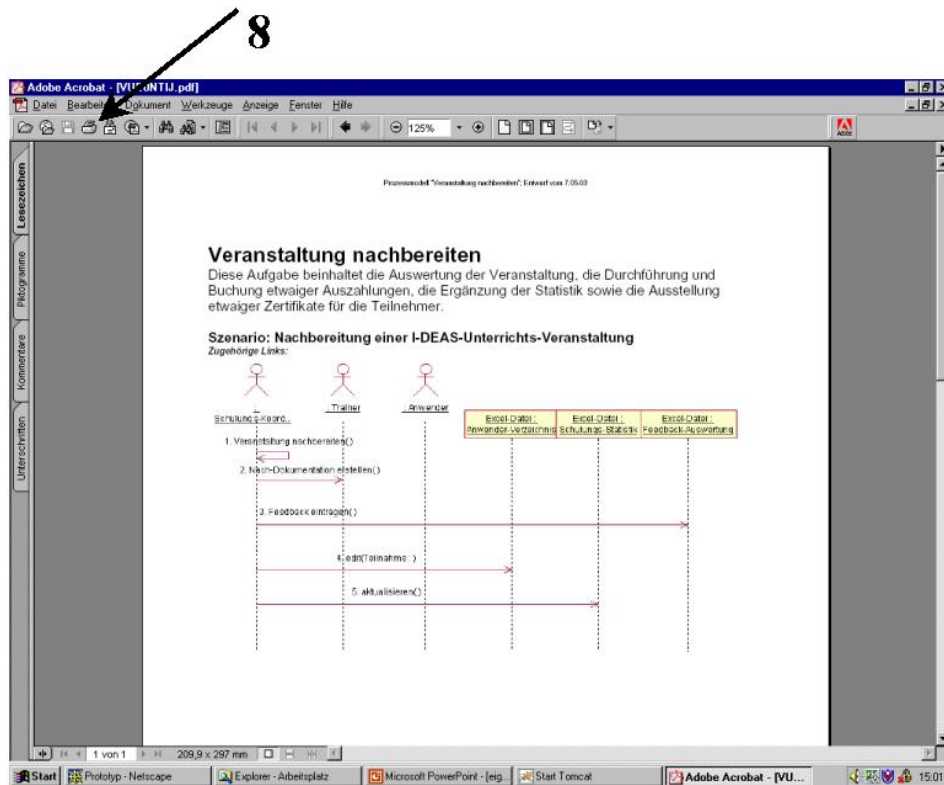


Abbildung 49: Szenario 1, Schritt 8

5.3.2 Szenario 2 : Optimierung des Prozesses „System pflegen“

Ausgangssituation:

Klaus P. sei Prozessverantwortlicher für Releasewechsel bei IPP. Bei den letzten Releasewechseln eines CAD-Systems wurden im Zuge der Anwendertests zahlreiche Wünsche für das Customizing des Systems gesammelt. Bisher konnten diese Wünsche nicht vollständig berücksichtigt werden, da der Prozess vor allem auf einen möglichst zügigen Release-Wechsel abgestimmt war. Jetzt soll dem Customizing im Prozess mehr Raum gegeben werden. Daher möchte er den Prozess „System pflegen“ gemeinsam mit den entsprechenden Mitarbeitern optimieren. Dazu verteilt er zunächst einen Ausdruck des Modells zur Vorbereitung eines Arbeitstreffens. Dann soll das Modell in der Arbeitssitzung anhand einer Beamer-Präsentation durchgespielt und diskutiert werden.

Umsetzung:

Er öffnet das Prozessmodell im Publikationssystem. In der Modellnavigation klickt er zunächst auf „Hauptaufgaben“ (1), worauf die einzelnen Hauptaufgaben als Unterpunkte erscheinen. Hier klickt er auf „System pflegen“ (2).

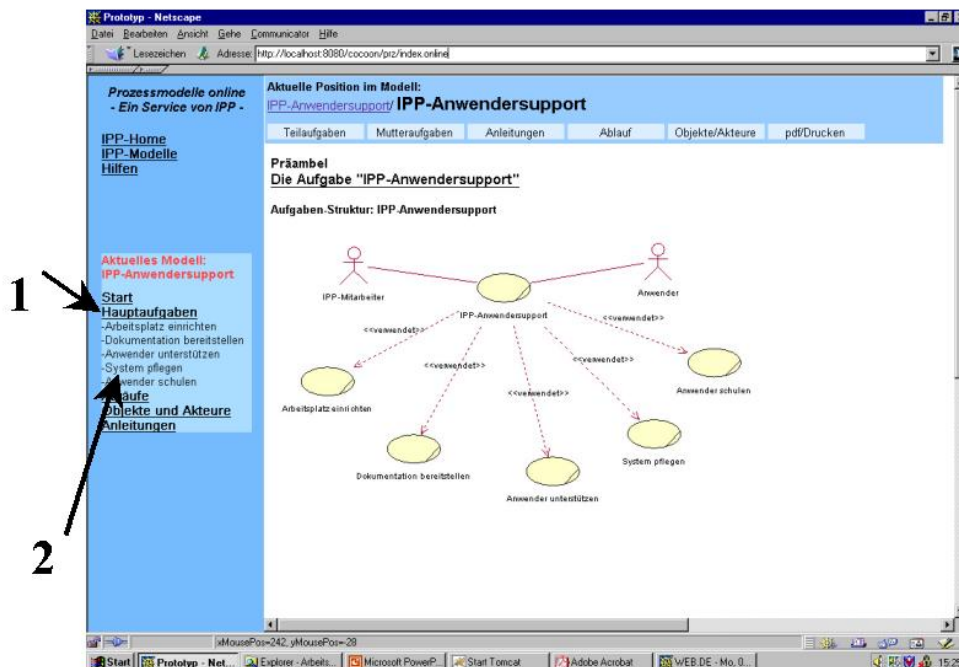


Abbildung 50: Szenario 2, Schritte 1 und 2

Es öffnet sich die Übersicht über die Teilaufgaben des Prozesses „System pflegen“. Er klickt auf den Button „pdf/Drucken“ (3) und wählt im Menü „Gesamt-Aufgabe“ (4) aus. Die Druckversion des Prozessmodells wird im pdf-Reader angezeigt und kann ausgedruckt werden (5).

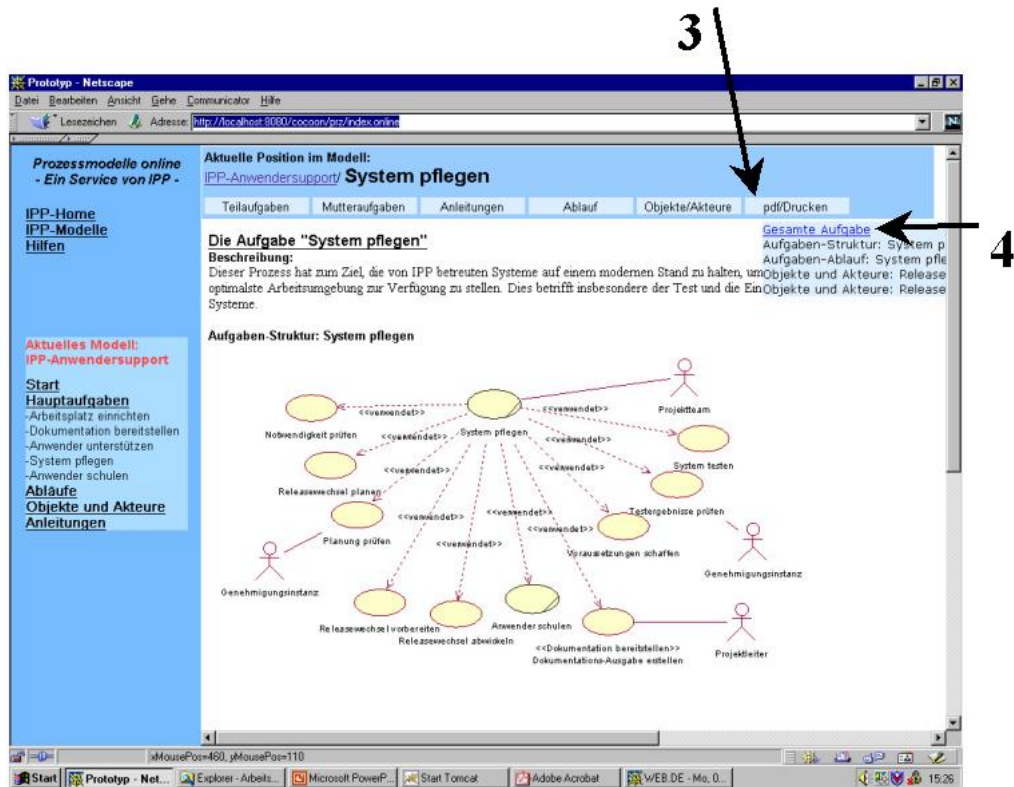


Abbildung 51: Szenario 2, Schritte 3 und 4

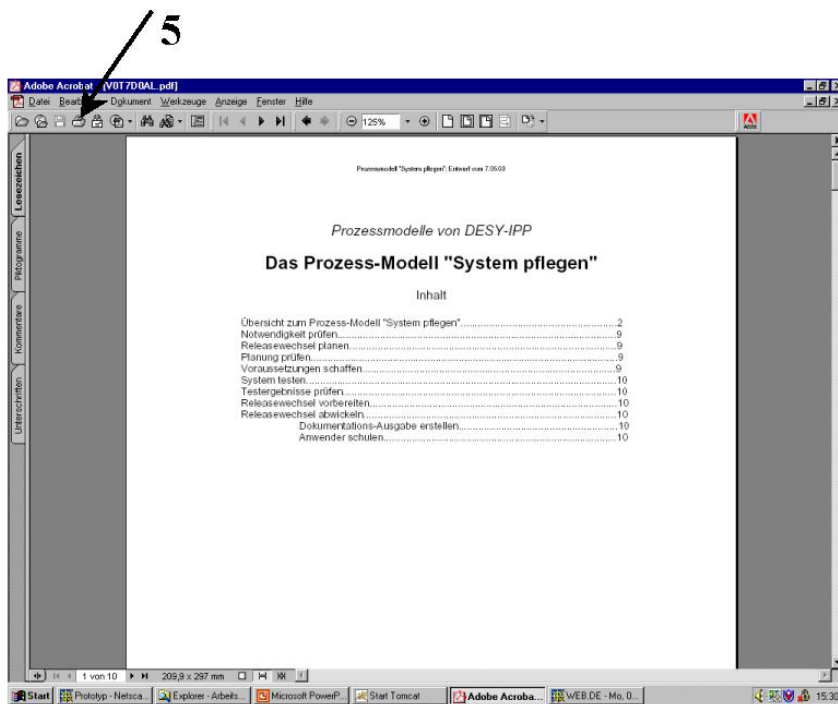


Abbildung 52: Szenario 2, Schritt 5

Während des Arbeitstreffens wird das Publikationssystem als Beamer-Projektion verwendet. Die einzelnen Aufgaben des Prozesses(6) werden anhand des Prozessablaufs durchgesprochen. Ein Teilnehmer hat eine Frage zu den beteiligten Akteuren. Daraufhin wird der Button „Objekte/Akteure“ betätigt (7) und im Menü „Release-Wechsel-Beteiligte“ angeklickt (8).

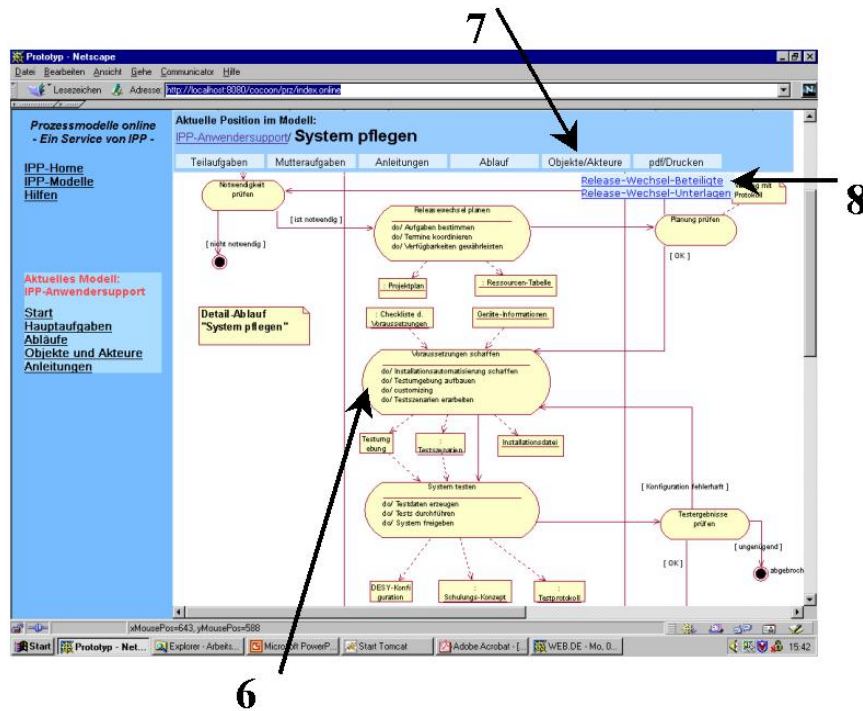


Abbildung 53: Szenario 2, Schritte 6 bis 8

Die verschiedenen Rollen werden mit den wesentlichen Beziehungen zu anderen Klassen angezeigt (9). Die weitere Diskussion behandelt die Struktur der Teilaufgaben. Dazu wird der Button „Teilaufgaben“ betätigt (10) und im Menü „Übersicht“ (11) aufgerufen.

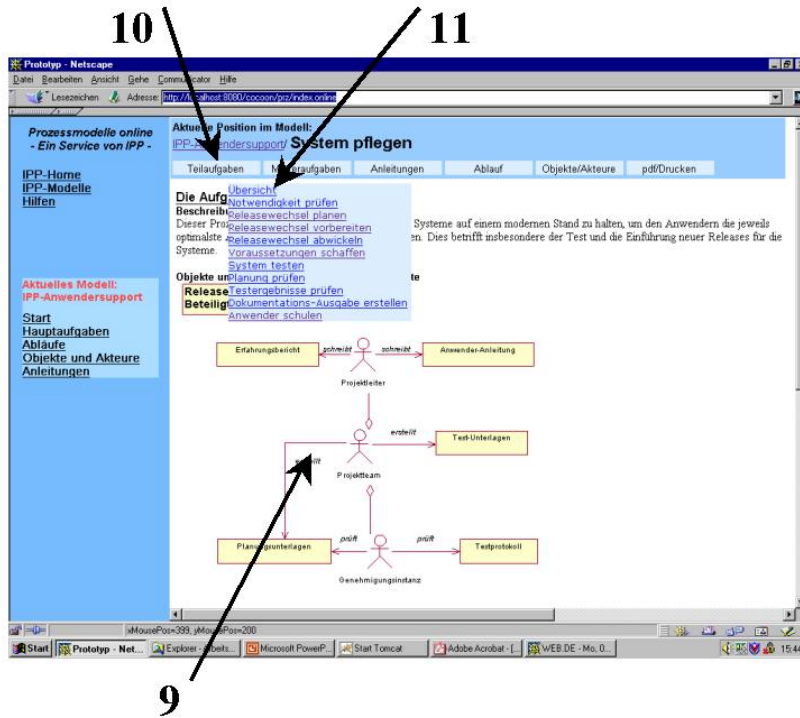


Abbildung 54: Szenario 2, Schritte 9 bis 11

Unter anderem soll die Vorbereitung des Releasewechsels optimiert werden. P. klickt auf das entsprechende Element (12). Das Kontextfenster zeigt die Kurzbeschreibung und den Link zum Öffnen der Aufgabe (13). [...]

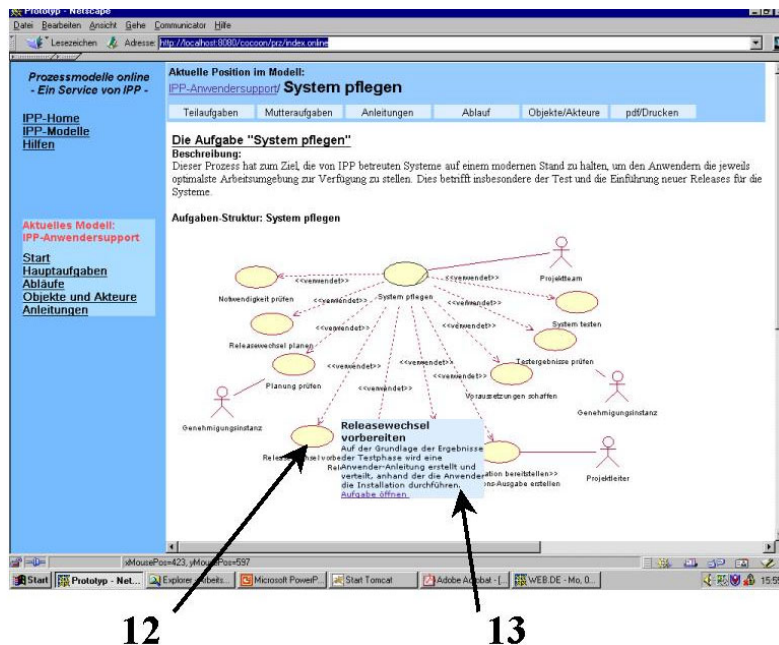


Abbildung 55: Szenario 2, Schritte 12 und 13

5.3.3 Szenario 3: Anfrage im Prozess „Arbeitsplatz einrichten“

Ausgangssituation:

Stefanie E. sei Konstrukteurin bei DESY und benötigt einen zweiten Rechner mit Zugriff auf das Engineering Data Management System. Von ihrem Gruppenleiter erfährt sie, dass IPP dafür zuständig sei und sie die Details im Intranet finde.

Umsetzung:

Sie geht auf die Website von IPP und findet einen Link auf die Beschreibung des Prozesses „Arbeitsplatz einrichten“. Der Link führt sie zum Ablauf des Prozesses. Sie erkennt die Aufgabe „Systemnutzung beantragen“ (1) als ersten Schritt des Ablaufes und klickt darauf, um Details zu erfahren. Das Kontextfenster zeigt ihr eine Kurzbeschreibung und einen Link zum Öffnen der Aufgabe (2), den sie betätigt.

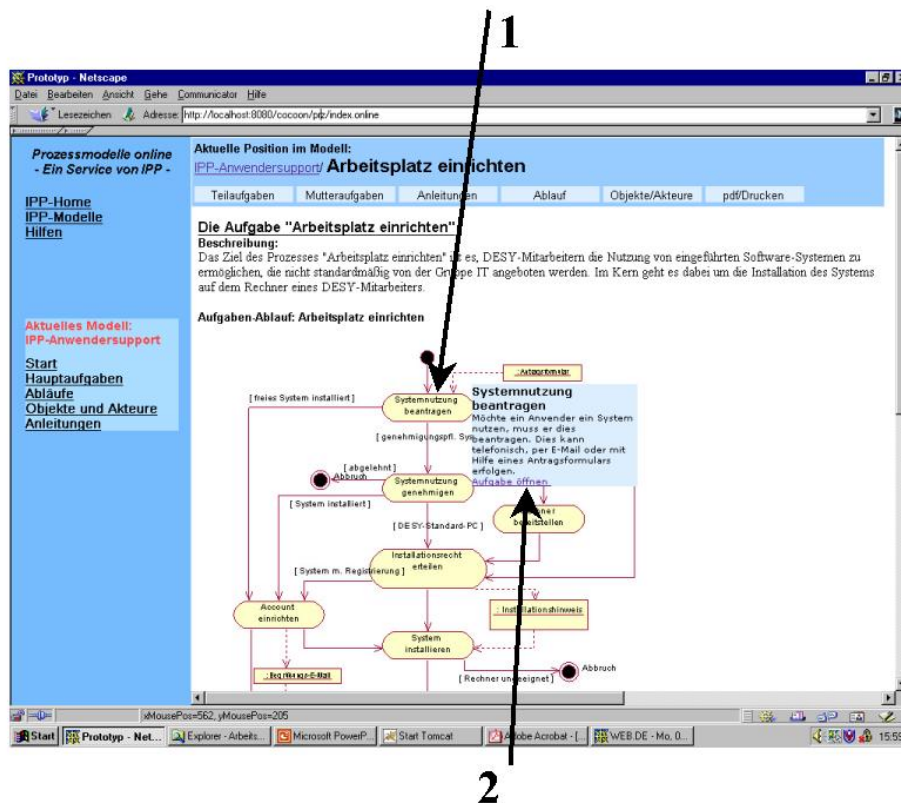


Abbildung 56: Szenario 3, Schritte 1 und 2

Sie gelangt zu den Anleitungen über die Anmeldung der verschiedenen Systeme, findet darunter auch EDMS und kann die einzelnen Schritte ablesen (3). Sie erkennt, dass sie zuerst ein Formular „Rechner anmelden“ ausfüllen muss (4), und klickt auf das entsprechende Element. Das erscheinende Kontextfenster enthält einen Link auf eine html-Seite, den sie betätigt (5).

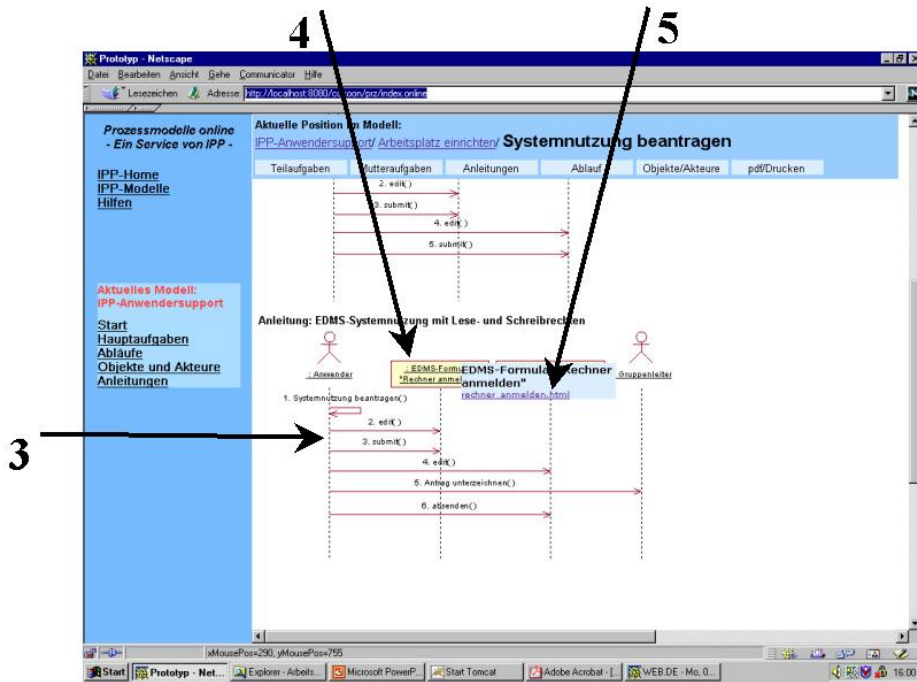


Abbildung 57: Szenario 3, Schritte 3 bis 5

Es öffnet sich ein neues Fenster mit dem EDMS-Formular „Rechner anmelden“. Hier kann sie ihre Daten eintragen (6). Anschließend arbeitet sie die Anleitung weiter ab.

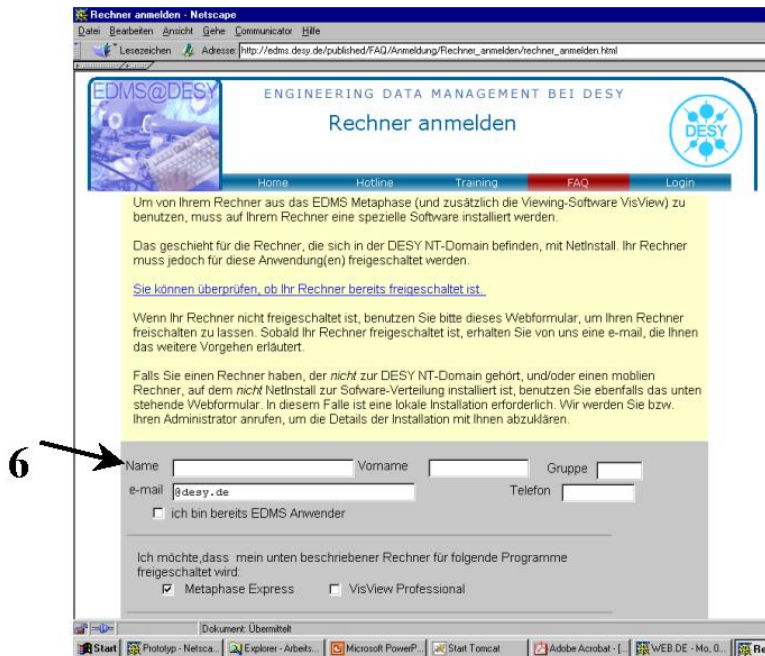


Abbildung 58: Szenario 3, Schritt 6

6 Datensicht und Systemsicht

Dieses Kapitel beschreibt die technischen Eigenschaften und Rahmenbedingungen des Publikationssystems. Sie sollen gewährleisten, dass das System mit vertretbarem Aufwand betrieben und konfiguriert werden kann. Zudem sollen die Daten der Prozessmodelle mit geringem Pflegeaufwand eingebunden und aktualisiert werden können.

Diese Anforderungen werden im Rahmen der folgenden Funktionen umgesetzt.

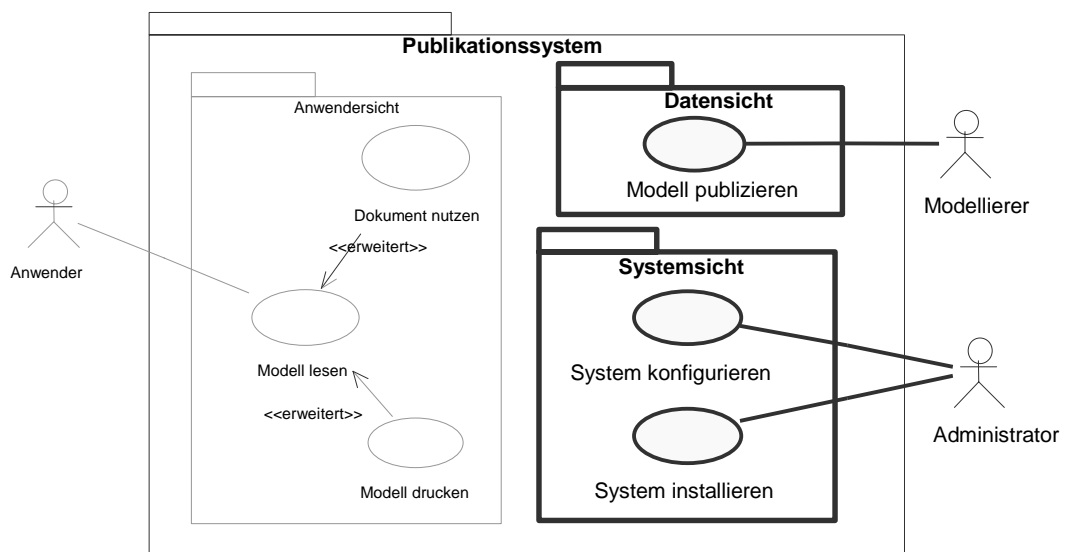


Abbildung 59: Funktionen der Datensicht und Systemsicht

Die Funktion "Modell publizieren" umfasst das Verfahren, mit dem ein Modell neu mit Hilfe des Systems publiziert wird oder die Daten eines bereits publizierten Modells aktualisiert werden.

Im Rahmen der Funktion "System konfigurieren" können die Datenorganisation und -transformation sowie die Gestaltung der Publikationen an veränderte Anforderungen angepasst werden. Dazu werden die entsprechenden Anwendungen und Schnittstellen bereitgestellt.

Die Funktion „System installieren“ umfasst die notwendigen Tätigkeiten bei der Neuinstallation des Systems in einem Client-Server-System.

6.1 Datensicht

Innerhalb der Datensicht werden der Datenexport aus dem Modellierungswerkzeug sowie die Strukturierung und das Format der Modelldaten im Publikationssystem bestimmt.

Wesentlicher Bestandteil der Datensicht ist die Funktion "Modell publizieren", die durch den Modellierer ausgeführt wird.

6.1.1 XML als Speicherformat der Modelldaten

Die Speicherung der Modelldaten im Publikationssystem muss einige grundlegende Anforderungen erfüllen:

- Konsistenz: Die ursprünglich in *Rational Rose* erstellte Modellstruktur muss innerhalb des Speicherformats abgebildet bzw. rekonstruiert werden können.
- Kompatibilität: Die technischen Voraussetzungen für die Speicherung der aus dem Modellierungswerkzeug (hier: *Rational Rose*) exportierten Daten im gewählten Format müssen vorhanden sein bzw. mit angemessenem Aufwand hergestellt werden können.
- Performanz: Die Daten müssen differenziert extrahiert werden können, um sie für die Generierung der Publikationen weiterzuverwenden.

Weiteres Kriterium für die Wahl des Speicherformats ist die Verfügbarkeit geeigneter Editoren und Verarbeitungssysteme.

Diese Anforderungen werden durch das Format XML (Extensible Markup Language) erfüllt. XML ist dabei, sich als Standard für die inhaltliche Strukturierung von Dokumenten durchzusetzen, insbesondere in Verbindung mit der elektronischen Publizierung und der Generierung verschiedener Publikationsformen aus einer Datenquelle (vgl. dazu Walsh 1998 und W3C 2003). Wesentliches Merkmal von XML-basierten Publikationen ist die getrennte Verwaltung von Inhalt, Struktur und Design. Dadurch kann ein einheitliches Erscheinungsbild aller Publikationen einer Organisation oder eines Anwendungsbereichs leicht hergestellt werden. Die Inhalte für alle Publikationen werden an zentraler Stelle in XML-Dateien aufbewahrt und können so effizient und konsistent aktualisiert werden.

Inhalte in XML-Dokumenten werden mit Hilfe eines hierarchischen Baumes von Elementen und Attributen strukturiert. Die Benennung der Elemente und Attribute ist durch den Standard nicht vorgegeben und kann daher entsprechend der jeweiligen

Informationsstruktur frei definiert und erweitert werden. Abbildung 60 zeigt eine prinzipielle XML-Struktur anhand von Personendaten.

```
<person geschlecht="P">
  <name>Müller</name>
  <vorname>Peter</vorname>
  <adresse>
    <strasse>Hauptstrasse</strasse>
    <hausnr>444</hausnr>
    <plz>10001</plz>
    <ort>Berlin</ort>
  </adresse>
  <telefon
    standort="privat">093741357</telefon>
  <telefon
    standort="mobil">084385734</telefon>
</person>
```

Abbildung 60: Beispiel für die Struktur von XML-Daten

Der Aufbau eines Elements wird in folgender Abbildung wiedergegeben:

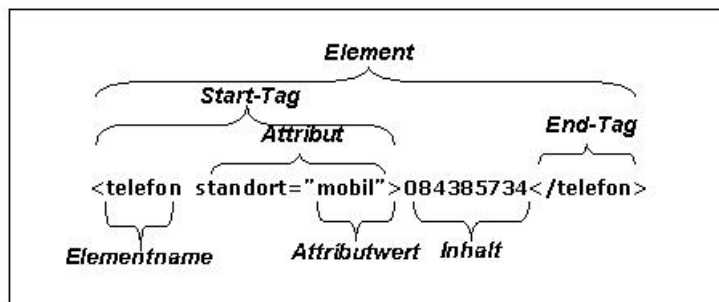


Abbildung 61: Aufbau eines XML-Elements

Der einheitliche Aufbau von XML-Dateien mit kongruent strukturierten Inhalten wird durch eine DTD-Datei oder ein XML-Schema definiert. Beide dienen der Validierung der XML-Dateien bei der Erstellung bzw. Aktualisierung und ermöglichen so eine standardisierte Verarbeitung der Daten (vgl. dazu Vonhoegen 2002).

Ein wesentlicher Vorteil von XML sind die bereits bestehenden Möglichkeiten, sowohl Online- als auch Print-Dokumente aus derselben Quelle heraus zu generieren. Mit Hilfe der XML- Namensräume XSLT und XSL-FO können in Verbindung mit geeigneten Parsern die gewünschten HTML- bzw. PDF-Dokumente generiert werden. Mit XPath steht zudem eine mächtige Abfragesprache zur Datenselektion zur Verfügung (s. Abschnitt 6.2).

Inzwischen ist das Angebot der Anwendungen und Prozessoren für die Erstellung und Verarbeitung XML-basierter Dokumente ausreichend groß und qualitativ hochwertig, so dass die Entwicklung eines Dokumentationssystem mit ihrer Hilfe zu einem stabilen und zuverlässigen Produkt führt.

6.1.2 Bewertung des Standards XMI

Die Durchsetzung von XML zeigt sich auch an der UML selbst. Verschiedene Software-Unternehmen haben sich 1999 im Rahmen der Object Management Group (OMG) auf den Standard XMI (XML Metadata Interchange) verständigt, mit dem Modell- und Metadaten in einem einheitlichen XML-Format textuell abgebildet werden. Dadurch werden folgende Zwecke erfüllt:

- UML-Modelle können zwischen verschiedenen Modellierungswerkzeugen ausgetauscht werden. Somit wird die Zusammenarbeit von Software-Entwicklern erleichtert.
- Metamodelle gemäß dem MOF-Standard können werkzeugunabhängig eingesetzt werden.
- XML-Schemata, DTDs und XML-Dokumente können aus UML-Modellen abgeleitet werden. Dies ist eine Voraussetzung, um Quellcode für XML-Anwendungen automatisch zu generieren.

(vgl. dazu Jeckle 2003 und OMG 2002a)

Ein Dokumentationsverfahren auf Basis von XMI kann somit bei entsprechender Anpassung universell eingesetzt werden, d.h. mit jedem Modellierungswerkzeug, das eine XMI-Schnittstelle besitzt.

Abbildung 62 zeigt einen kleinen Auszug aus der XMI-Fassung des UML-Modells des IPP-Anwendersupports, die mit Hilfe des Add-Ins von *Unisys* aus *Rational Rose* (vgl. dazu Jeckle 2003a) exportiert wurde. Daran wird deutlich, dass mit XMI wesentlich mehr Daten exportiert werden, als bei Umsetzung der vorgeschlagenen Modellierungskonventionen für das Publikationssystem benötigt werden.

Zwar berücksichtigt der eigentliche XMI-Standard bisher keine grafischen Daten (vgl. OMG 2001, S. 22), mit der Verabschiedung der Version 2.0 der UML im Juni 2003 ist jedoch eine Integration grafischer Daten in XMI in die Wege geleitet worden (vgl. Oesterreich 2003, S. 36).

Unisys bietet für *Rational Rose* schon jetzt eine eigene XMI-Erweiterung als Teil seines Export-Tools an (vgl. Jeckle 2003a).

```

<UML:UseCase xmi.id="S.174.1141.54.54" name="Anfrage zuteilen" visibility="public" isSpecification="false" isRoot="false"
  isLeaf="false" isAbstract="false" namespace="S.174.1141.54.36">
  <UML:Namespace.ownedElement>
    <UML:Collaboration xmi.id="S.174.1141.54.55" name="Anfrage zuteilen-Collaboration" visibility="public"
      isSpecification="false" isRoot="false" isLeaf="false" isAbstract="false">
      <UML:Namespace.ownedElement>
        <UML:ClassifierRole xmi.id="G.559" name="" visibility="public" isSpecification="false" isRoot="false"
          isAbstract="false" base="S.174.1141.53.44" availableFeature="S.174.1141.53.65
          53.50" message2="G.563 G.564 G.565" message1="G.563 G.565">
          <UML:ClassifierRole.multiplicity>
            <UML:Multiplicity>
              <UML:Multiplicity.range>
                <UML:MultiplicityRange low="1" high="1" />
              </UML:Multiplicity.range>
            </UML:Multiplicity>
          </UML:ClassifierRole.multiplicity>
        </UML:ClassifierRole>
        [... ca. 40 weitere Elemente]
        <UML:Collaboration.interaction>
          <UML:Interaction xmi.id="G.566" name="{Use Case View::Anwender schulen::Anfrage zuteilen}Anfrage
            zuteilen" visibility="public" isSpecification="false">
            <UML:Interaction.message>
              <UML:Message xmi.id="G.563" name="Anfrage zuteilen()" visibility="public"
                isSpecification="false" sender="G.559" receiver="G.559" message3="G.564"
                communicationConnection="G.560" action="XX.24.1142.0.764"/>
              <UML:Message xmi.id="G.564" name="editTeilnahme()" visibility="public"
                isSpecification="false" sender="G.559" receiver="G.562" message3="G.565"
                predecessor="G.563" communicationConnection="G.561" action="XX.24.1142.0.765"/>
              <UML:Message xmi.id="G.565" name="Anmeldung best&#x00e4;tigen()" visibility="public"
                isSpecification="false" sender="G.559" receiver="G.559" predecessor="G.564"
                communicationConnection="G.560" action="XX.24.1142.0.766"/>
            </UML:Interaction.message>
          </UML:Interaction>
        </UML:Collaboration.interaction>
      </UML:Collaboration>
    </UML:UseCase>
    [... ca. 20 weitere Elemente]
    -----
    <UML:TaggedValue xmi.id="XX.24.1142.0.745" tag="persistence" value="transient" modelElement="G.562"/>
    <UML:TaggedValue xmi.id="XX.24.1142.0.757" tag="documentation" value="Ein potentieller Teilnehmer, der sich durch eine §
    wird einer Veranstaltung zugeteilt, die gerade entsprechend seinem Bedarf vorbereitet wird." modelElement="S.174.1141.54.54" />
    <UML:TaggedValue xmi.id="XX.24.1142.0.759" tag="persistence" value="transient" modelElement="G.562"/>
    <UML:TaggedValue xmi.id="XX.24.1142.1.767" tag="documentation" value="Das Ziel des Prozesses "Arbeitsplatz einrichten"
    Nutzung von eingere&#x00e4;hrten Software-Systemen zu erm&#x00e4;glichen, die nicht standardm&#x00e4;ßig von de
  
```

Verwendung zahlreicher Attribute zur genauen Spezifikation

Über Diagramme werden gemäß dem Standard nur semantische, aber keine grafischen Informationen gespeichert (Erweiterung von Unisys ist nicht abgebildet).

Die semantische Beschreibung eines Elements wird vollständig wiedergegeben.

Dokumentationen werden als „tagged values“ separat von den Elementen abgelegt

Abbildung 62: Modelldaten gemäß dem Standard XMI 1.1 (Auszug)

Der Standard XMI ist geeignet für eine vollständige textuelle Beschreibung der semantischen Modellstruktur. Für die Verwendung in diesem Publikationssystem ist er jedoch trotz der Erweiterung mit grafischen Daten mit einigen Nachteilen verbunden.

- Die große Menge an Daten, die nicht verwertet werden, belegt Speicherplatz und könnte die Geschwindigkeit der Transformation vermindern.
- Die Verwendung zahlreicher Attribute und die Trennung zwischen Elementen und ihren „Tagged Values“ macht es komplizierter, Inhalte zu adressieren und zu extrahieren.
- Die Komplexität von XMI erschwert die Pflege und Konfiguration des Systems sowie die Einarbeitung neuer Administratoren.

- Die grafischen Erweiterungen von *Unisys* für *Rational Rose* enthalten keine Koordinaten für verweissensitive Bereiche.
- *Unisys* exportiert keine Ränge von Anwendungsfällen.

6.1.3 Entwicklung einer proprietären XML-Lösung

Diese Nachteile bestehen nicht, wenn die Daten auf anderem Wege exportiert und gemäß einem eigens neu definierten XML-Schema gespeichert werden.

Das Dokumentationswerkzeug *SODA for Word* bietet dazu ausreichende Möglichkeiten (vgl. dazu IBM 2003).

Die XML-Datenquelle wird dabei mit Hilfe eines *SODA*-Templates erstellt, mit dem ausgewählte Modellelemente und -Beschreibungen aus der *Rational-Rose*-Modelldatei extrahiert und in XML-Tags eingebettet werden. Um eine modellunabhängige Verwendung zu ermöglichen, ist eine einheitliche Ablagestruktur der Modellelemente innerhalb der Modell-Datei notwendig. Gleichzeitig muss das Template den Vorgaben des XML-Schemas entsprechen, das Grundlage der Transformationen im Publikationssystem ist. Beides wird durch die Beachtung der vorgeschlagenen Modellierungskonventionen gewährleistet.

Rational-SODA ermöglicht zwar den Export der Modelldiagramme als OLE-Objekte, jedoch nicht den Export von grafischen Daten, die XML-fähig sind (z.B. in Form von Vektorkoordinaten). Möglich ist jedoch der Export der Diagrammattribute einschließlich der Diagramm-ID und der Koordinaten für die Definition verweissensitiver Bereiche in HTML.

Die UML-Diagramme selbst müssen separat exportiert werden. Dies kann mit Hilfe des Web-Publishers von *Rational Rose* in effizienter Weise durchgeführt werden. Dabei werden die Diagramme unter Verwendung der Diagramm-IDs in systematischer Weise in einem selbstgewählten Ordner abgelegt. Die in der XML-Datenquelle enthaltenen Daten über die Diagramme können dann verwendet werden, um auf die Diagrammdateien zuzugreifen und sie in die Publikationen zu integrieren.

Mit einem proprietären XML-Schema kann auf dieser Grundlage eine Datenstruktur definiert werden, die genau auf den Anwendungsbedarf des Publikationssystems zugeschnitten ist (s. Anhänge B und C)

Dieser Lösungsweg wurde für den Prototyp des Publikationssystems realisiert.

Abbildung 63 zeigt dazu den Auszug einer entsprechenden XML-Datenquelle, der das gleiche Element zeigt wie der XMI-Auszug in Abbildung 62.

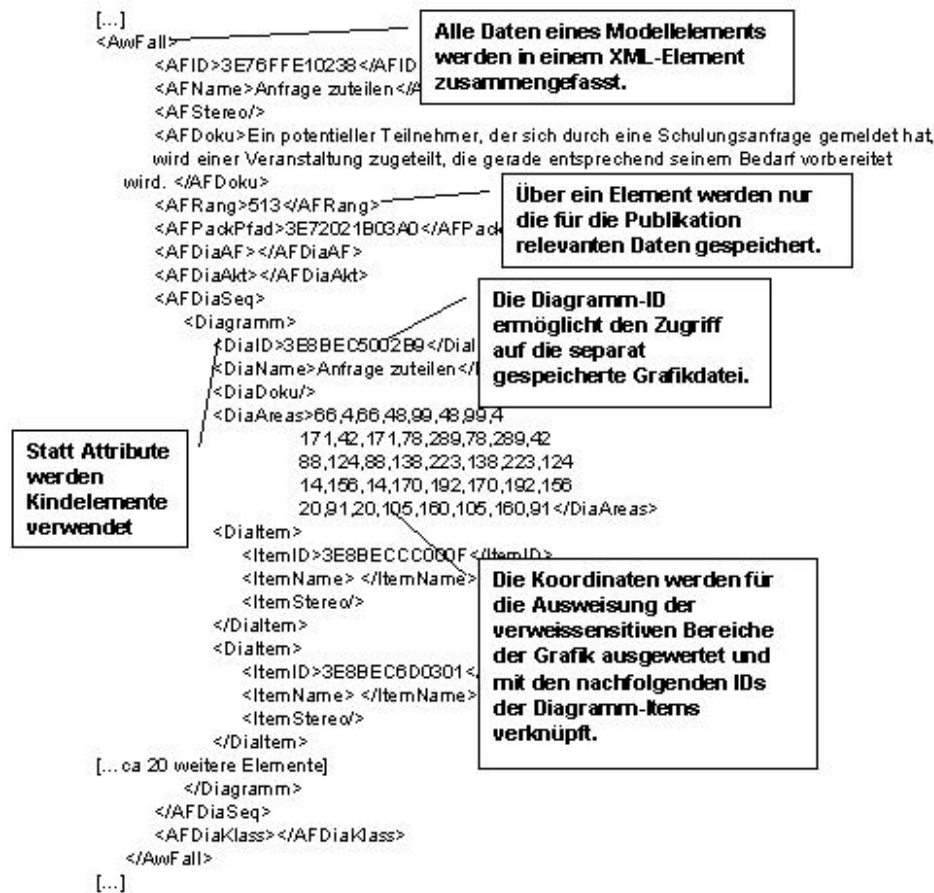


Abbildung 63: Modelldaten gemäß dem proprietären XML-Schema (Auszug)

Die Datenmenge ist gegenüber dem Unisys-Export etwa im Verhältnis 1:6 geringer (gemessen an der Dateigröße). Die Inhaltstruktur bildet dabei die Ablagestruktur der Elemente und Diagramme in *Rational Rose* ab, d.h. die Diagrammdaten sind direkt als Kindelemente bestimmten Anwendungsfällen zugeordnet. Somit kann der Zugriff auf die Inhalte einfacher realisiert werden.

6.1.4 Die Funktion „Modell publizieren“

Der folgende Ablauf zeigt im Überblick die Aufgaben bei der Generierung der Datenquellen im Rahmen der Funktion "Modell publizieren". Auslöser des Prozesses ist das Ende eines Modellierungsvorgangs in *Rational Rose*. Es werden zwei Fälle unterschieden, einerseits die erstmalige Publikation eines Modells, andererseits die Aktualisierung eines bereits publizierten Modells.

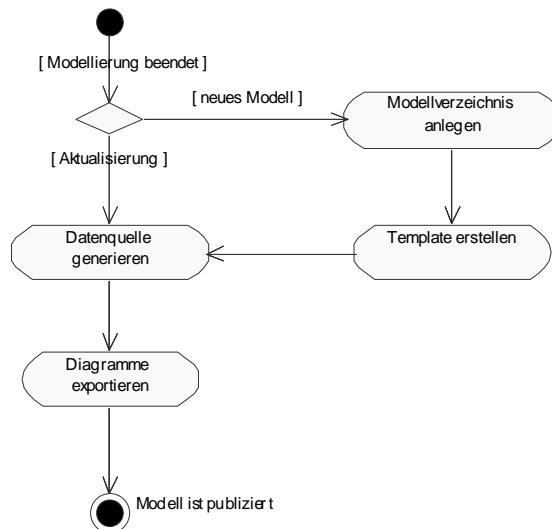


Abbildung 64: Ablaufstruktur der Funktion „Modell publizieren“

Beschreibung der einzelnen Aufgaben:

- Modellverzeichnis anlegen – Innerhalb des Systemverzeichnisses wird ein Modellverzeichnis angelegt, das die XML-Datenquelle, die Diagramme und die eingebundenen Dokumente enthalten wird. Name und Kurzbeschreibung des neuen Modells werden in der Modell-Liste des Systems eingetragen.
- Template erstellen – Für den Export der Daten aus *Rational Rose* und die Generierung der XML-Datei wird ein *SODA*-Modell-Template erstellt. Vorlage für das Modell-Template ist das Meta-Template.
- Datenquelle generieren - Die XML-Datei wird mit Hilfe des *SODA*-Modell-Template generiert und im Modellverzeichnis gespeichert.
- Diagramme exportieren – Mit Hilfe des Web-Publishers von *Rational Rose* werden die Diagramme als jpeg-Grafiken in das Modellverzeichnis exportiert.

Die Struktur der an der Datengenerierung beteiligten Objekte wird durch folgendes Diagramm wiedergegeben:

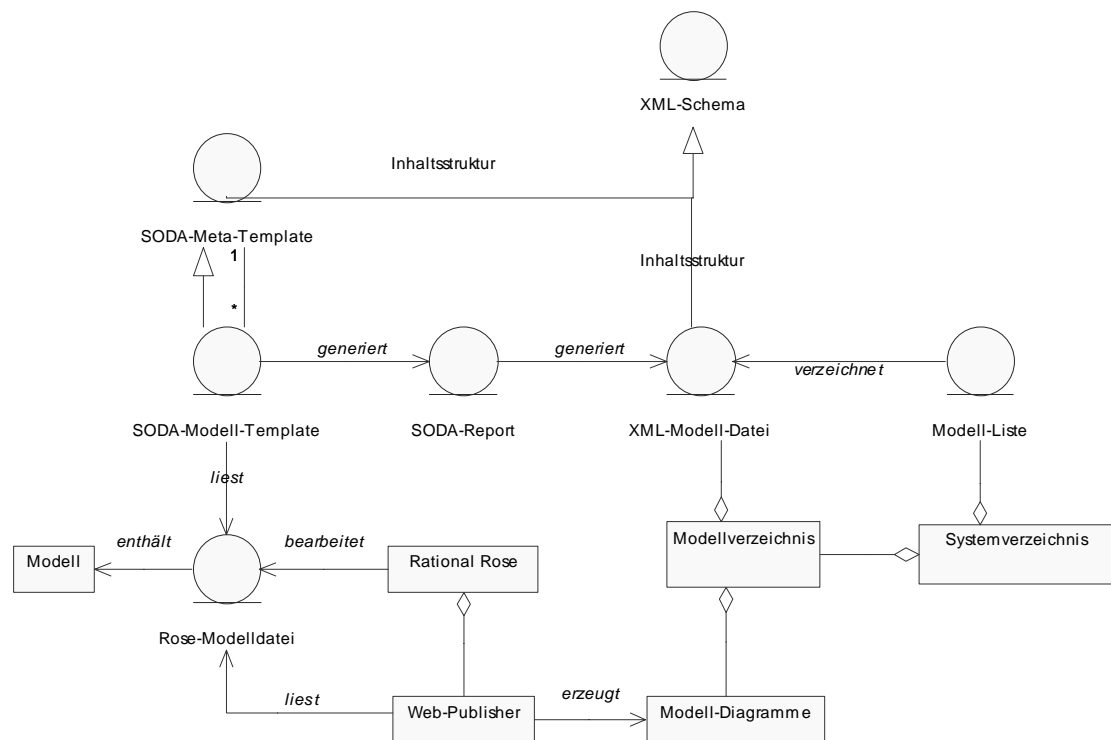


Abbildung 65: Objektstruktur der Datengenerierung

Beschreibung der dargestellten Klassen:

- Modell – das zu publizierende Prozessmodell
- Modelldiagramme – die zu publizierenden UML-Diagramme
- Modell-Liste – verzeichnet Namen und Kurzbeschreibungen der publizierten Modelle
- Modellverzeichnis – enthält die XML-Modelldatei die Modelldiagramme und im Modell eingebundene Dateien
- Rational Rose – Modellierungssoftware
- Rose-Modelldatei – Datei mit den Modelldaten im Rational-Rose-Format
- SODA-Meta-Template – dient als Vorlage für alle Modell-Templates und vererbt an diese die Inhaltsstruktur.
- SODA-Modell-Template – SODA-Template eines Modells für den Export aus *Rational Rose* und die Generierung der XML-Modelldatei.
- SODA-Report – temporäres Format der Modelldaten vor der Speicherung im XML-Format

- Systemverzeichnis – enthält neben den Modellverzeichnissen die Systemdateien
- Web-Publisher – Add-In-Werkzeug von *Rational Rose* für den Export der Modelldaten im HTML-Format
- XML-Modell-Datei – enthält die Modelldaten im XML-Format gemäß dem XML-Schema
- XML-Schema – definiert die Inhaltsstruktur der XML-Modelldatei und somit auch der Templates.

Im Rahmen der Systemdokumentation werden dem Modellierer Handlungsanleitungen in Form von UML-Sequenzdiagrammen bereitgestellt, die die detaillierte Umsetzung jeder Aufgabe der Funktion „Modell publizieren“ beschreiben.

6.1.5 Szenario 4 : Aktualisierung eines Modells

Das folgende Szenario verdeutlicht die praktische Umsetzung der Funktion bei der Aktualisierung von Modelldaten.

Ausgangssituation:

Klaus P. habe als Prozessverantwortlicher den Prozess „System pflegen“ überarbeitet. Nun möchte er die neue Version im Publikationssystem veröffentlichen.

Umsetzung:

Er öffnet das zugehörige *SODA*-Template, mit dem die zu publizierenden Daten aus der Modelldatei exportiert und in das XML-Format überführt werden. Im Menü „SODA“ klickt er auf „Generate Report“ (1), um den Datenexport auszuführen.

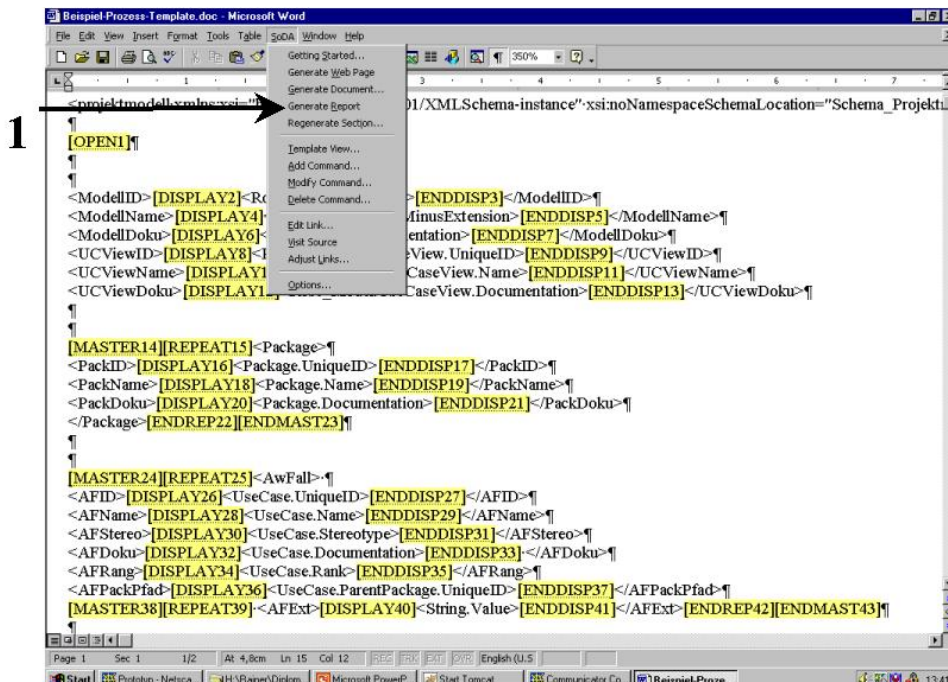


Abbildung 66 : Szenario 4, Schritt 1

Den so erzeugten Bericht speichert er als XML-Datei (2) im entsprechenden Modellordner des Publikationssystems (3) ab.

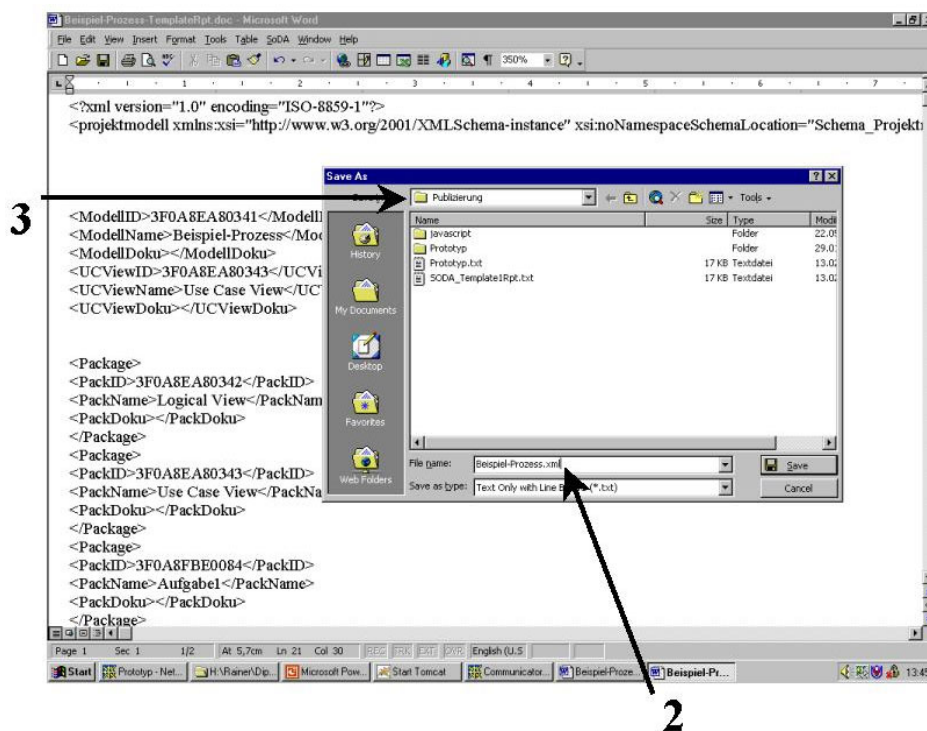


Abbildung 67: Szenario 4, Schritte 2 und 3

Anschließend wechselt er zum Prozessmodell in *Rational Rose* um die Diagramme zu exportieren. Dazu wählt er im Menü „Tools“ (4) den Web Publisher. Er benennt den Modellordner als Ablageverzeichnis der Diagramme (6) und wählt JPEG als Grafikformat (7). Durch Klicken auf „Publish“ (7) löst er den Exportvorgang aus.

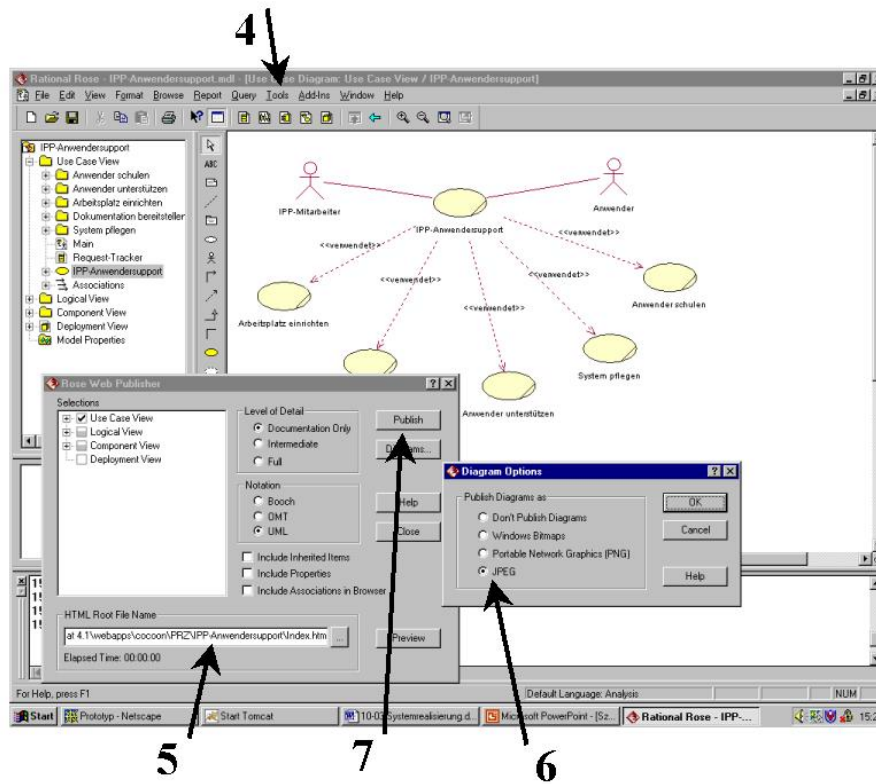


Abbildung 68: Szenario 4, Schritte 4 bis 7

6.2 Systemsicht

Die Systemsicht des Publikationssystems umfasst die Transformation der Modelldaten aus der Datenquelle in geeignete Publikationsformate für die Intranetdarstellung und den Ausdruck. Im Rahmen der Funktion "System konfigurieren" ändert der Administrator den Aufbau und die Darstellung der Publikationen.

6.2.1 Codierung der Publikationsstruktur und -gestaltung

Bei der Transformation der Modelldaten im XML-Format kommen drei bedeutende XML-Standards zur Anwendung, die zusammen die XSL (Extensible Stylesheet Language) bilden (vgl. dazu Vonhoegen 2002).

Mit XSLT (XSL-Transformations) werden Templates erstellt, anhand derer die Element-Hierarchie der XML-Quelle ausgewertet wird. Ein Template legt fest, welche Elemente und Attribute in das Ausgabedokument übernommen werden. Gleichzeitig können unabhängig von der Datenquelle Zeichen hinzugefügt werden, die im Ausgabedokument dargestellt bzw. ausgewertet werden. Dadurch können sowohl Text als auch neue XML-Elemente eingefügt und mit den Daten der XML-Datei verbunden werden. Da HTML den Strukturregeln von XML entspricht, können mit XSLT auch HTML-Dokumente erzeugt werden.

XSL-FO (XSL-Formatting Objects) wird zusammen mit XSLT eingesetzt. Mit ihr werden die Strukturelemente einer Printfassung festgelegt und formatiert. Ebenso wie HTML können auch XSL-FO-Elemente zusammen mit den über XSLT selektierten Modelldaten ausgegeben werden. Voraussetzung für die Generierung eines Printdokuments auf diesem Weg ist die Verwendung eines entsprechenden Parsers.

XPath ist das entscheidende Mittel für die Selektion der Modelldaten. Mit ihm ist ein exakter Zugriff auf Elemente, Elementgruppen und Attribute der XML-Datenquelle sowie deren Inhalte möglich. XPath-Ausdrücke werden in bestimmten Attributen von XSLT-Elementen verwendet, um die Verarbeitung der Elementhierarchie der XML-Datei zu steuern und die Inhalte des Ausgabedokuments zu bestimmen

Das Design des Publikationssystems wird über CSS-Stylesheets bestimmt. Dynamische Funktionen insbesondere bei der Navigation wurden mit Javascript-Funktionen umgesetzt. Beides ist in separaten Dateien abgelegt, die mit den XSL-Dateien verknüpft sind.

6.2.2 Apache Cocoon als Transformationswerkzeug

Die Transformation von XML-Daten kann prinzipiell auf zwei Wegen erfolgen:

1. Die XML-Datei wird zusammen mit einer XSL-Datei an den Client-Browser übermittelt, der die Transformation selbst durchführt.
2. Die Transformation wird serverseitig durchgeführt und nur die Daten des Ausgabedokuments (HTML oder PDF) selbst werden an den Client-Browser übermittelt, um sie dort anzuzeigen.

Da die Verarbeitung von XML- mit XSL-Dateien längst noch nicht zum Standard der gängigen Browser gehört, wäre die erste Variante mit zu vielen Hürden für den Anwender bei der Nutzung des Systems verbunden.

Demgegenüber stehen für die serverseitige Transformation zahlreiche Prozessoren zur Verfügung.

Für den Prototyp des hier entwickelten Publikationssystems fiel die Entscheidung auf *Cocoon 2* der *Apache Software Foundation*. *Cocoon* ist als „Opensource“ entwickelt worden und bietet ein umfassendes XML-Publikations-Werkzeug. Das wesentliche Merkmal von *Cocoon* ist die vollständig separate Definition von Inhalt, Struktur, Design und Management eines webbasierten Publikationssystems.

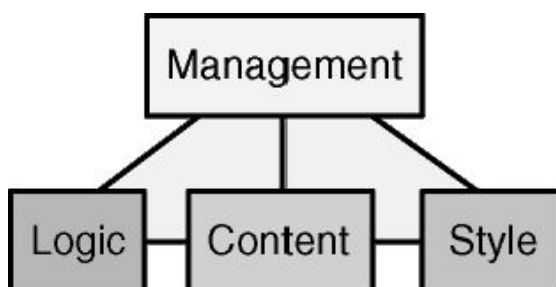


Abbildung 69: Die vier Betrachtungs-Ebenen eines Webangebots bei Cocoon (Apache 2002)

Das folgende Diagramm verdeutlicht die Funktion der verschiedenen Objekte bei *Cocoon*. Zentrales Organisations-Mittel für die Transformationen sind die *Matcher*. Durch sie wird die Verbindung zwischen URI-Anfrage des Browsers, Datenquelle, XSL-Datei und dem Ausgabeformat definiert.

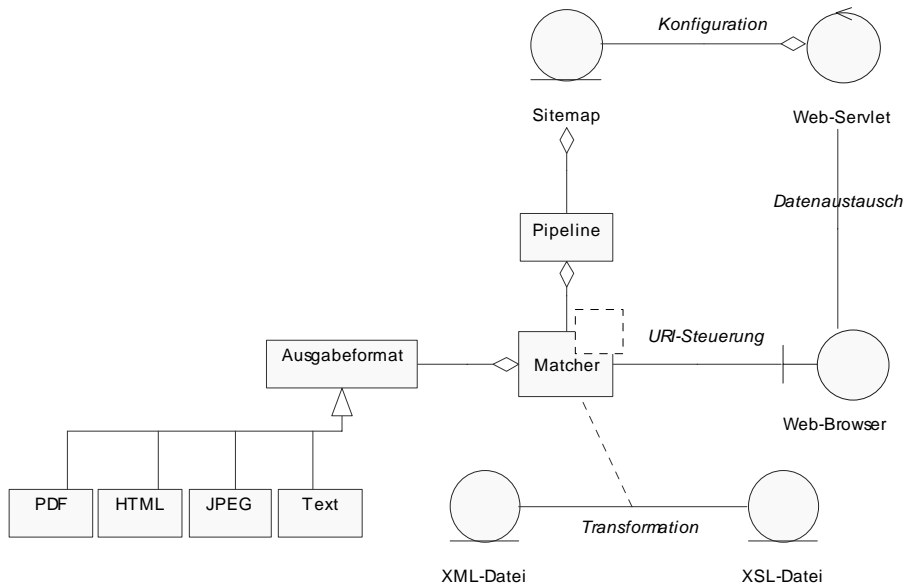


Abbildung 70: Objektstruktur der Daten-Transformation

Die Kommunikationsschritte der verschiedenen Objekte in *Cocoon* werden in folgendem Szenario verdeutlicht. Eine URI-Anfrage über den *Web-Browser* des Clients löst demnach eine bestimmte Transformation aus, wenn dafür ein entsprechender *Matcher* definiert ist.

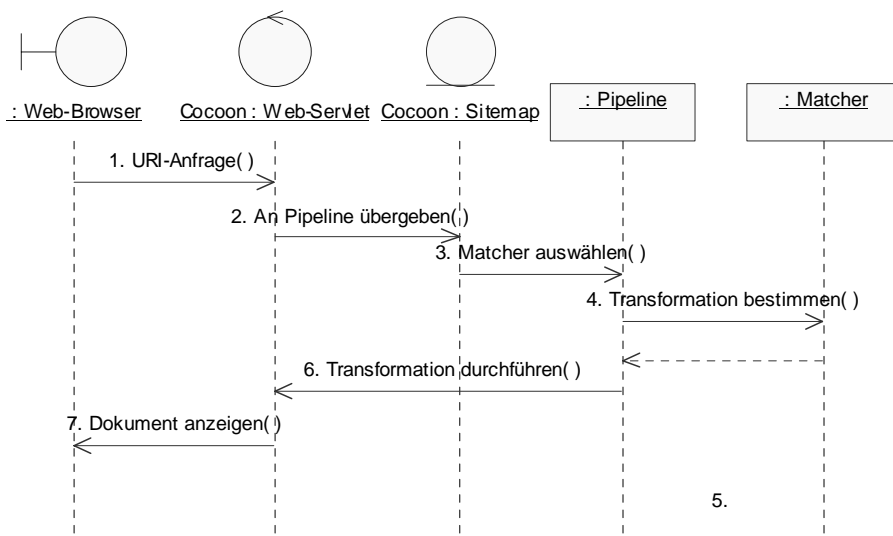


Abbildung 71: Szenario der Verarbeitung einer URI-Anfrage in Cocoon

6.2.3 Szenario 5: Funktion "System konfigurieren" - Erweiterung der Modellinhalte

Dieses Szenario zeigt beispielhaft eine mögliche Aufgabe, die im Rahmen der Funktion „System konfigurieren“ zu erfüllen ist.

Ausgangssituation:

Die Modellierungskonventionen sehen bisher keine Verwendung von dokumentierten Packages vor. Diese werden jedoch für ein neues Prozessmodells einer anderen DESY-Abteilung benötigt. Die Modellierungskonvention werden entsprechend erweitert. Der Systemadministrator erhält den Auftrag, auch das Publikationssystem entsprechend anzupassen.

Umsetzung:

Zunächst erweitert er das XML-Schema, mit dem die XML-Datenquellen der Modelle validiert werden. Dazu definiert er einen komplexen Elementtypen (1), der die Struktur der Packagedaten beschreibt. Ein Element dieses Typs verankert er als Kindelement (2) des äußersten Elements.

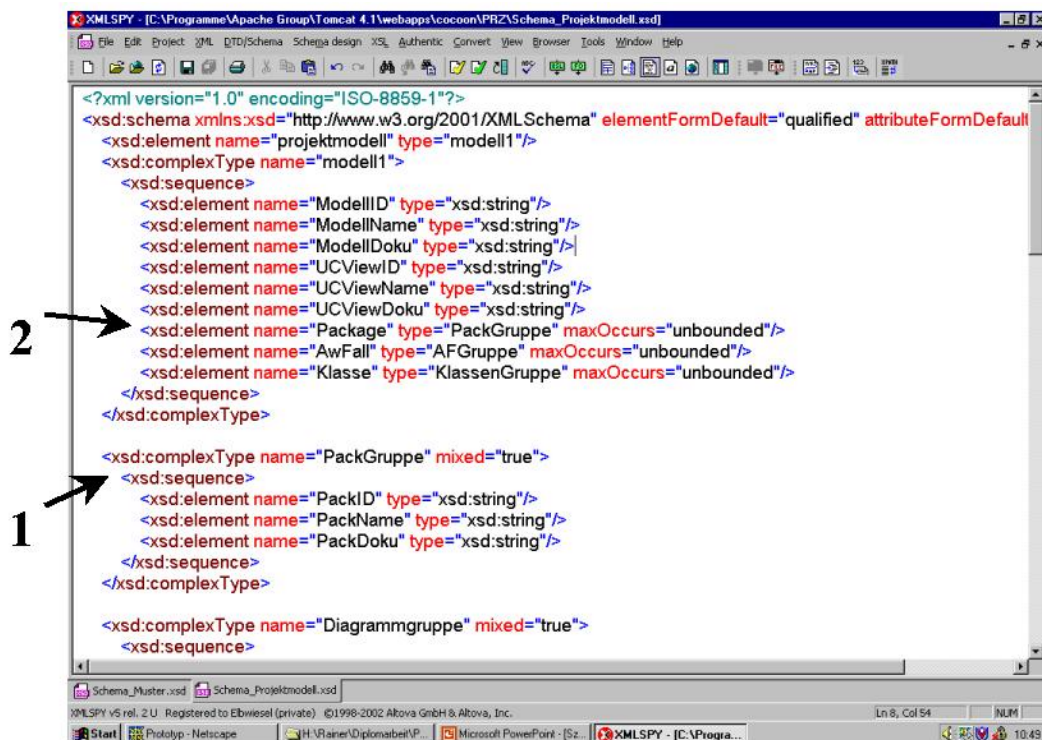


Abbildung 72: Szenario 5, Schritte 1 und 2

In den SODA-Templates der Modelle fügt er mit Hilfe der Funktion „Add Command“ (3) zusätzliche Felder für die Package-Daten ein (4).

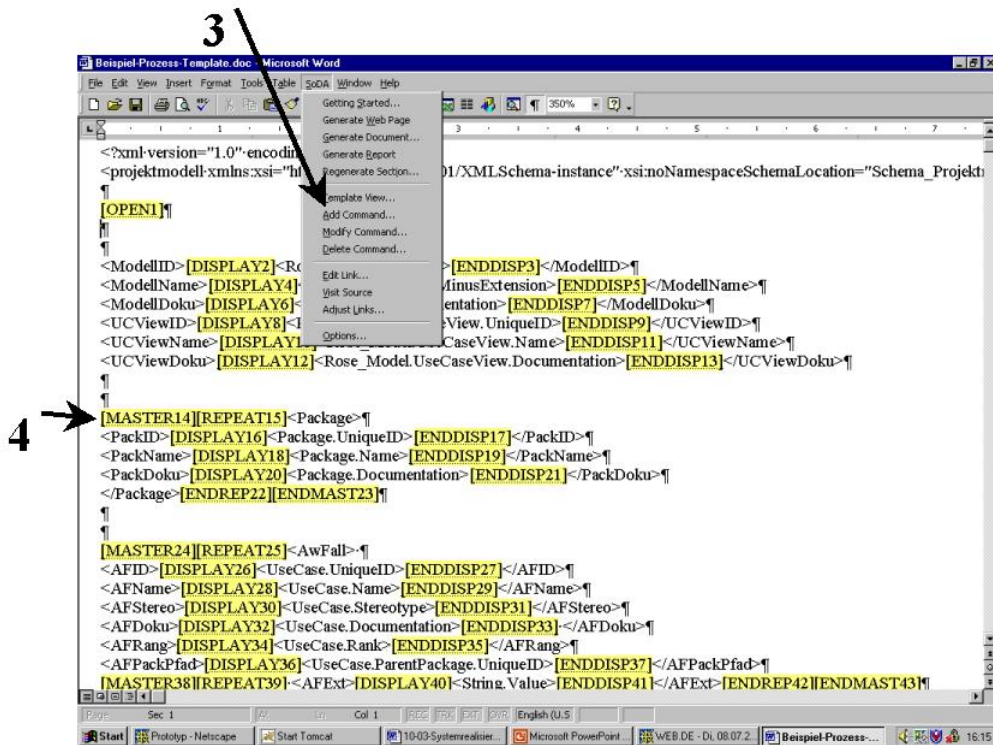


Abbildung 73: Szenario 5, Schritte 3 und 4

In der XSL-Datei für die Diagramm-Darstellung im Intranet fügt er die Daten der Packages in die Layer ein, die die Kontext-Beschreibungen der zugehörigen Diagrammelemente enthalten (5). Eine ähnliche Ergänzung nimmt er in der XSL-Datei für die Generierung der Druckversion vor.

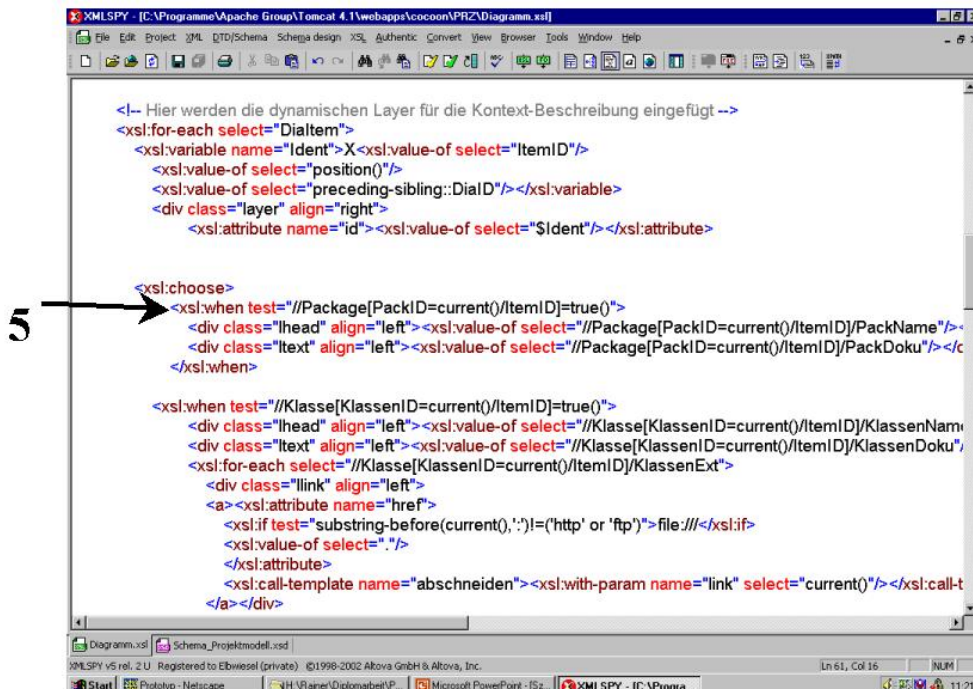


Abbildung 74: Szenario 5, Schritt 5

Im Ergebnis erscheinen im Publikationssystem Kontextbeschreibungen auch nach dem dem Anklicken von Package-Symbolen (6).

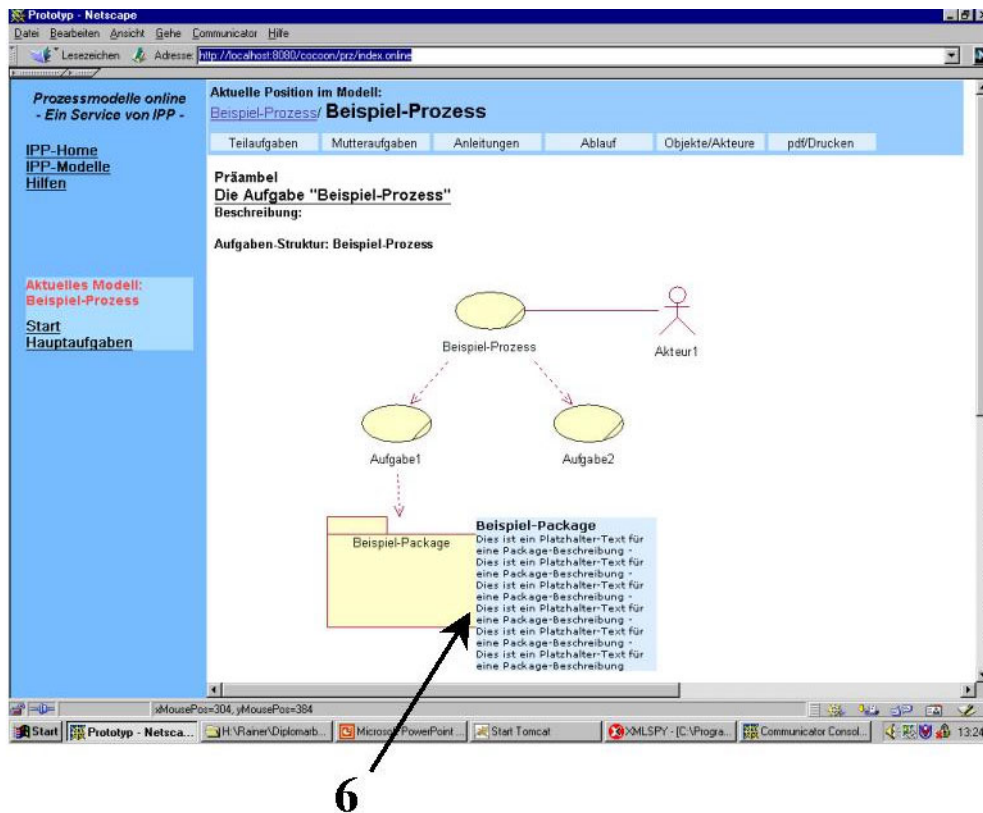


Abbildung 75: Szenario 5, Schritt 6

6.2.4 Checkliste zur Funktion „System installieren“

Hardware:

- Windows Server in einem Client-Server-Netzwerk

Systemvoraussetzungen beim Client:

- Installation eines HTML-Browsers : Netscape 4.7 , Internet Explorer 5.0+ oder Mozilla
- Installation eines PDF-Readers

Erforderliche Applikationen:

- Für die Modellierung : *Rational Rose 2000 Enterprise Edition*
- Für den Datenexport: *Rational SODA for Word* und *MS Word*
- Für die Erstellung des XML-Schemas und der XSL-Dateien : XML-Editor (z.B. *XML-Spy*)
- Für die Transformation und Webanbindung (auf dem Server) : Apache Cocoon 2.04. unter Apache Tomcat 4.1.X

Installation der Verzeichnisse und Dateien:

- Erstellung des Systemverzeichnisses auf dem Server
- Kopieren folgender Dateien in das Systemverzeichnis :
 - System-Sitemap
 - XML-Schema
 - *SODA*-Meta-Template
 - XSL-Dateien
 - CSS- und Javascript-Dateien.
- Konfiguration der System-Sitemap für die URI-Steuerung und das Datei-Mapping
- Konfiguration der Cocoon-Sitemap mit Verweis auf die System-Sitemap
- Definition eines Ordnersystems für die einzubindenden Dokumente sowie des Zugriffs darauf.

7 Schlussbetrachtungen

7.1 Ergebnisüberblick

Das Ergebnis dieser Arbeit ist ein System zur Dokumentation von Geschäftsprozessmodellen. Die zu Beginn formulierten Ziele (s. Abschnitt 1.2) sind in folgender Weise umgesetzt worden:

- Es wurden Modellierungskonventionen entworfen, die die verwendete Notation der Modellierungssprache UML in ihrer Anwendung vereinfachen und vereinheitlichen, insbesondere hinsichtlich der Modellstruktur und der Gestaltung von Szenarien, die als Handlungsanweisungen dienen sollen.
- Es wurde ein Publikationssystem entwickelt, das Dokumentationen von UML-Prozessmodellen sowohl in interaktiv navigierbarer als auch in linearer, ausdrückbarer Form generiert. Die Anwendersicht des Systems spiegelt die Modellierungskonventionen wieder und ermöglicht eine intuitive Nutzung mit praktischem Wert (Anleitungen, Tätigkeitsprofile, Dokumente etc.).
- Hilfen und Arbeitsmaterialien werden über Verweise zu den entsprechenden Dokumenten in die Modelle integriert und im Publikationssystem den Nutzern in Form von Hyperlinks zugänglich gemacht.
- Um den Publikationsvorgang effizient ausführen zu können, wurde die Abfolge der manuellen Bedienungsschritte definiert und dokumentiert. Sie sind gleichzeitig die Basis für eine mögliche programmiertechnische Automatisierung.
- Das System ist für mehrere Modelle nutzbar. Voraussetzung ist, dass sie einer einheitlichen Struktur folgen, die mit dem definierten proprietären XML-Schema abgebildet werden kann. Auch diesem Zweck dient die Vereinheitlichung durch die vorgeschlagenen Modellierungskonventionen.

Auf dieser Grundlage bilden Publikationssystem und Modellierungskonventionen zusammen ein einsatzfähiges Werkzeug für die organisationsweite Dokumentation, Vermittlung und Umsetzung von Geschäftsprozessmodellen.

Mit Hilfe der Erfahrungen bei der täglichen Anwendung können System und Konventionen schrittweise optimiert und erweitert werden. Einige Optionen für einen breiteren und effizienteren Einsatz des Publikationssystems sind bereits im Laufe der Entwicklung deutlich geworden (s.u.).

7.2 Weiterführende Aspekte

7.2.1 *Werkzeugunabhängiger Einsatz mittels XMI*

Der Einsatz des Publikationssystems ist gegenwärtig allein auf die gemeinsame Verwendung mit *Rational Rose* abgestimmt. Soll das System werkzeugunabhängig eingesetzt werden, existieren zwei Lösungswege mit sehr unterschiedlicher Reichweite. Zum einen kann jedes im Standard XMI 1.x vorliegende Modell in *Rational Rose* importiert, dort an die Modellierungskonventionen angepasst und schließlich wie beschrieben exportiert und publiziert werden.

Der zweite Weg beinhaltet die Definition einer eigenen XMI-Schnittstelle für das Publikationssystem. Dies erfordert jedoch voraussichtlich eine vollständige Neukonzeption der Datenebene des Systems, da XMI 1.x keine Informationen über die Diagramme eines Modells liefert (vgl. OMG 2001 S. 22), sondern nur über die semantischen Beziehungen der Modellelemente. Zwar soll die kurz vor Abschluss dieser Arbeit erfolgte vollständige Verabschiedung von UML 2.0 einen verbesserten Diagrammaustausch mittels XMI ermöglichen (vgl. dazu Oestereich 2003), es ist jedoch noch nicht bekannt, ob dabei auch die Ablagestruktur der Diagramme innerhalb des Repositories des Modellierungswerkzeugs abgebildet wird. Die in dieser Arbeit vorgeschlagene Struktur der Modelle bezieht aber die Diagramme als wesentliches Gliederungsmittel mit ein.

Auf jeden Fall böte die Integration grafischer Daten in XMI die Möglichkeit, einen separaten Diagrammexport zu umgehen. Dazu könnten die Diagramme beispielsweise über einen SVG-Parser generiert werden, entweder einmalig während des Publikationsvorgangs oder in Echtzeit bei Abfrage des Diagramms durch den Client-Browser.

7.2.2 *Verknüpfung mit Workflow-Systemen*

Durch die Integration von Verweisen auf externe Arbeitsmaterialien konnte bereits ein praktischer Nutzen des Publikationssystems für den betrieblichen Alltag implementiert werden. Ein nächster Schritt kann darin bestehen, die Modelldaten mit den im Alltag verwendeten Workflow-Systemen konsequent zu verknüpfen. Eine derartige Integration könnte aus folgenden Maßnahmen bestehen:

- Übernahme der Prozess-Benennungen aus dem Modell in die Benutzeroberfläche des Workflow-Systems;

- Unterstützung der Vorgänge durch Ticket-Vorlagen, die die definierten Prozessabläufe widerspiegeln;
- Aufbau einer bi-direktionalen Verweisstruktur zwischen Workflow-System und der Modellpublikation;
- Übernahme der Verweise auf externe Arbeitsmaterialien.

Die Abbildung der Modelldaten mit Hilfe eines einheitlichen XML-Schemas bietet dabei Möglichkeiten, die über die reine Publikation des Modells hinaus gehen. Durch ihre standardisierte Struktur und Verarbeitung sind XML-Dateien universell in unterschiedlichen technischen Zusammenhängen einsetzbar. Daher sollte geprüft werden, inwiefern die XML-Dateien der Modelle gleichzeitig als Datenquellen für gegenwärtige oder zukünftige workflow-unterstützende Systeme dienen können, um die beschriebene Integration zu erreichen.

7.2.3 Geschäftsprozessmanagement ist Informationsmanagement

Im Rahmen dieser Arbeit wurden die vielfältigen Anforderungen deutlich, die das betriebliche Informationsmanagement im Rahmen der Geschäftsprozessmodellierung erfüllen kann.

Es genügt nicht, dass das Informationsmanagement nur als eine Aufgabe der technischen Umsetzung bei der Prozessdokumentation oder der Entwicklung von Workflow-Systemen betrachtet wird. Vielmehr muss es bereits im Vorfeld als Katalysator tätig werden, um ein Geschäftsprozessmodell passend für die Zielgruppe und die jeweilige Verwendung aufzubereiten und zu vereinheitlichen. Die dabei erarbeiteten Modellierungskonventionen bilden den Rahmen für die weitere Modellierung. Je nach Anforderungen und Erfahrungen im Laufe der kontinuierlichen Verbesserung der publizierten Modelle können die Konventionen dann schrittweise modifiziert und erweitert werden.

Daher sollten Organisationen, die ihre Geschäftsprozesse neu definieren oder optimieren, ihre für das Informationsmanagement verantwortlichen Mitarbeiter in allen Phasen der Geschäftsprozessmodellierung kontinuierlich mit einbeziehen.

Quellenverzeichnis

- Apache 2002 Apache: *Understanding Apache Cocoon*. 2002. Online im Internet:
URI: <http://cocoon.apache.org/2.0/userdocs/concepts/index.html>
[Stand 2003-06-19]
- Becker 2000 Becker, Jörg (Hrsg.) ; Kugler, Martin (Hrsg.); Rosemann, Michael (Hrsg.) : *Prozessmanagement : ein Leitfaden zur prozessorientierten Organisationsgestaltung* . 2. Aufl.. Berlin : Springer, 2000 . - ISBN 3-540-67197-8
- Booch 1999 Booch, Grady ; Rumbaugh, James; Jacobson, Ivar : *The Unified Modeling Language User Guide* . Boston : Addison-Wesley, 1999 . - ISBN 0201571684
- Bucher 2001 Bucher, Michael [u.a.]; Bullinger, Hans-Jörg (Hrsg.) ; IAO - Fraunhofer Institut für Arbeitswirtschaft und Organisation (Hrsg.): *Knowledge meets Process : Wissen und Prozesse managen im Intranet* . Stuttgart, 2001 . - ISBN 3-8167-5620-4
- Castela 2001 Castela, Nuno; Tribolet, Jose E.; Silva, Alberto; Guerra, Arminda : *Business Process Modeling with UML* . IN: *ICEIS 2001, Proceedings of the 3rd International Conference on Enterprise Information Systems, Setubal, Portugal, July 7-10, 2001. Volume 2* ., 2001, S. 679-685. Online im Internet: URI: <http://berlin.inesc.pt/alb/papers/2001/iceis2001-BusinessModelingUML.pdf> [Stand 2003-07-01]
- DESY 2000 DESY - Deutsches Elektronen Synchrotron (Hrsg.): *Über DESY : Kurzportrait*. 01.03.2000. Online im Internet: URI: <http://www.desy.de/html/ueberdesy/kurzportrait.html> [Stand 2003-07-04]
- DESY 2001 DESY - Deutsches Elektronen Synchrotron (Hrsg.): *Die Organisation des zentralen CAD-Supports am DESY* Version 0.3. Hamburg, 2001 Interner Bericht
- DESY 2003 DESY - Deutsches Elektronen Synchrotron (Hrsg.): *Wissenschaftlicher Jahresbericht 2002* . Hamburg, 2003 . Online im Internet: URI: <http://www.desy.de/f/jb2002/desy.pdf> [Stand 2003-07-15]
- Dumas 2001 Dumas, Marlon; Ter Hofstede, H.M. Arthur : *UML Activity Diagrams as a Workflow Specification Language* . IN: *Proc. of the International Conference on the Unified Modeling Language (UML). Toronto, Canada, October 2001* .Heidelberg : Springer, 2001. Online im Internet: URI: http://sky.fit.qut.edu.au/~dumas/uml01_dumas.pdf [Stand 2003-07-01]
- Ehlert 2002 Ehlert, Sven TU Berlin: *Einführung in die Software-Ergonomie* . Berlin, 2002. Online im Internet: URI: http://flp.cs.tu-berlin.de/~oseidel/Lehre/SS_2002/einfuehrung_in_die_softwareergonomie.pdf [Stand 2003-06-17]

- Eriksson 2000 Eriksson, Hans-Erik; Penker, Magnus : *Business Modeling with the UML : Business Patterns at Work* . New York : Wiley, 2000. - ISBN 0-471-29551-5
- Füerermann 1997 Füerermann, Timo; Dammasch, Carsten : *Prozessmanagement : Anleitung zur ständigen Verbesserung aller Prozesse im Unternehmen* . München : Hanser, 1997 (Pocket Power). - ISBN 3-446-19006-6
- Gadatsch 2002 Gadatsch, Andreas : *Management von Geschäftsprozessen : Methoden und Werkzeuge für die IT-Praxis : Eine Einführung für Studenten und Praktiker* . 2. Aufl.. Braunschweig : Vieweg, 2002. - ISBN 3-528-15759
- Geis 2002 Geis, Thomas : *Software-Ergonomie/Usability : Relevanz für Hersteller und Betreiber von Software* . , 2002. Online im Internet: URI: <http://www.redtenbacher.de/swergo/swergo.pdf> [Stand 2003-06-17]
- Goesmann 1998 Goesmann, Thomas; Föcker, Egbert; Striemer, Rüdiger ; ISST - Fraunhofer Institut Software- und Systemtechnik (Hrsg.): *Wissensmanagement im Kontext der Gestaltung und Durchführung von Geschäftsprozessen* . Dortmund, 1998 . Online im Internet: URI: http://www.isst.fhg.de/german/veroeffentlichungen/pdf_dateien/Ber48.pdf [Stand 2003-05-30]
- Grässle 2002 Grässle, Patrick; Baumann, Henriette; Baumann, Philippe : *UML projektorientiert* . 1. Aufl.. Bonn : Galileo, 2002 . - ISBN 3-89842-336-0
- Hartwig 2003 Hartwig, Ronald : *Software- Ergonomie Information* . Sereetz, 1999 . Online im Internet: URI: <http://www.rhaug.de/information/info.pdf> [Stand 2003-06-17]
- Herget 1997 Herget, Josef : Informationsmanagement . IN: Buder, Marianne [u.a.] (Hrsg.) : *Grundlagen der praktischen Information und Dokumentation, Band 2* . 4. Aufl. München : Saur, 1997. - ISBN 3-598-11309-9, S. 781-794
- Hruby 1998 Hruby, Pavel : *Structuring Specification of Business Systems with UML (with an Emphasis on Workflow Management Systems) (OOPSLA'98 Business Object Workshop IV)*. 1998. Online im Internet: URI: <http://www.jeffsutherland.org/oopsla98/pavel.html> [Stand 2003-07-01]
- Jeckle 2000 Jeckle, Mario ; Daimler Chrysler: *Objektorientierter Modellaustausch und XML-Spracherzeugung mit XMI* . Ulm, 2000 . Online im Internet: URI: <http://www.jeckle.de/files/oop2001.pdf> [Stand 2003-01-23]
- Jeckle 2003 Jeckle, Mario : *XML Metadata Interchange*. 24.06.2003. Online im Internet: URI: <http://www.jeckle.de/xmi.htm> [Stand 2003-06-25]

- Jeckle 2003a Jeckle, Mario : *XMI-Repräsentation mit Rational Rose Erweiterungen (XMI v1.0)*. 25.03.2003. Online im Internet: URI: http://www.jeckle.de/xmi_ex1.html#xmi10rose [Stand 2003-07-25]
- Lappe 2002 Lappe, Kathrin ; DESY - Deutsches Elektronen Synchrotron (Hrsg.): *Projektplanung und -steuerung bei der Erstellung von Multimedia-CD-ROMs durch objektorientierte Geschäftsprozessmodellierung* . Hamburg : DESY, 2002 (DESY-Thesis 2002-037). - ISSN 1435-8085
- Niedermaier 2003 Niedermaier, Stephan : *Cocoon Tutorial* . München : Logabit Netservices, 2003 . Online im Internet: URI: <http://www.pgmttools.de/downloader.php?id=11> [Stand 2003-07-01]
- Norm 1995 NORM DIN ISO 9241-11 : *Anforderungen an die Gebrauchstauglichkeit – Leitsätze* , 1995
- Norm 1996 NORM DIN ISO 9241-10 : *Grundsätze der Dialoggestaltung* , 1996
- Norm 1997 NORM DIN ISO 8402 : *Qualitätsmanagement und Qualitätssicherung - Begriffe* , 1997
- Oestereich 2003 Oestereich, Bernd; Weilkiens, Tim : UML 2.0 : Alles wird gut ? . IN: *Objektspektrum* . Troisdorf: Sigs-Datcom, (2003) H.1, S. 36-38. Online im Internet: URI: http://www.sigs.de/publications/os/2003/01/oestereich_OS_01_03.pdf [Stand 2003-06-23]
- OMG 2001 OMG - Object Management Group (Hrsg.): *UML 2.0 RFP Request For Proposal : UML 2.0 Diagram Interchange RFP* . Needham, MA, 2001 . Online im Internet: URI: <http://www.omg.org/docs/ad/01-02-39.pdf> [Stand 2003-06-23]
- OMG 2002 OMG - Object Management Group: *Introduction to OMG's Unified Modeling Language™ (UML™)*. 22.08.2002. Online im Internet: URI: http://www.omg.org/gettingstarted/what_is_uml.htm [Stand 2003-06-11]
- OMG 2003 OMG - Object Management Group (Hrsg.): *OMG Unified Modeling Language Specification* . Version 1.5. Needham, MA, 2003 . Online im Internet: URI: <http://www.omg.org/cgi-bin/doc?formal/03-03-01.pdf> [Stand 2003-06-03]
- OMG 2003a OMG - Object Management Group (Hrsg.): *XML Metadata Interchange (XMI) Specification : Version 2.0* . Needham, MA, 2003 . Online im Internet: URI: <http://www.omg.org/docs/formal/03-05-02.pdf> [Stand 2003-06-23]
- Osterloh 1999 Osterloh, Margit; Wübker, Sigrid : *Wettbewerbsfähiger durch Prozeß- und Wissensmanagement* . Wiesbaden : Gabler, 1999 . - ISBN 3-409-18981-5

- Rational 2001 Rational: *Business Modeling with the UML and Rational Suite AnalystStudio* , 2001 . Online im Internet: URI: http://www.rational.com/media/uml/resources/documentation/27070_BusinessModeling.pdf [Stand 2003-07-03]
- IBM 2003 IBM: *Rational SoDA: Product Information : Automating the creation and maintenance of project documentation*. IBM, 2003. Online im Internet: URI: <http://www.rational.com/products/soda/prodinfo.jsp> [Stand 2003-07-25]
- Scheer 1998 Scheer, A.-W. : *ARIS : Vom Geschäftsprozess zum Anwendungssystem* . 3. Aufl. Berlin : Springer, 1998 . - ISBN 3-540-63835-0
- Schmidt 2002 Schmidt, Günter : *Prozessmanagement : Modelle und Methoden* . 2. Aufl. Berlin : Springer, 2002 . - ISBN 3-540-43170-5
- Schneider 2000 Schneider, Wolfgang ; Gesellschaft Arbeit und Ergonomie - online e.V. (Hrsg.): *Grundsätze der Dialoggestaltung (ISO 9241-10)*. 03.02.2000. Online im Internet: URI: <http://www.sozialnetz.de/awca/pq.asp?id=luw> [Stand 2003-06-17]
- Schönsleben 2001 Schönsleben, Paul : *Integrales Informationsmanagement : Informationssysteme für Geschäftsprozesse - Management, Modellierung, Lebenszyklus und Technologie* . 2. Aufl. Berlin : Springer, 2001 . - ISBN 3-540-41712-5
- Specht 2000 Specht, Olaf; Schmitt, Ulrich : *Betriebswirtschaft für Ingenieure und Informatiker* . 5.Aufl. München : Oldenbourg, 2000 . - ISBN 3-486-25550-9
- Vonhoegen 2002 Vonhoegen, Helmut : *Einstieg in XML* . Bonn : Galileo Press, 2002 . - ISBN 3-89842-137-6
- W3C 2003 W3C - World Wide Web Consortium (Hrsg.): *Extensible Markup Language (XML)*. 10.06.2003. Online im Internet: URI: <http://www.w3.org/XML/> [Stand 2003-06-25]
- Walsh 1998 Walsh, Norman : *What is XML ?*. 03.10.1998. Online im Internet: URI: <http://www.xml.com/pub/a/98/10/guide1.html> [Stand 2003-06-25]
- Werewka 2002 Werewka, Oskar ; DESY - Deutsches Elektronen Synchrotron (Hrsg.): *Design and Implementation of an XML-Based Call-tracking System for Monitoring and Improving Business-Process Performance* . Hamburg : DESY, 2002 (DESY-Thesis 2002-038). - ISSN 1435-8085
- Wöhe 2000 Wöhe, Günter; Döring, Ulrich : *Einführung in die Allgemeine Betriebswirtschaftslehre* . 20. Aufl. München : Vahlen, 2000 . - ISBN 3-8006-2550-4
- Zimmermann 2002 Zimmermann, Silvia : *Was ist Usability ?*. 09.12.2002. Online im Internet: URI: http://www.usability.ch/News_D/usanov.htm [Stand 2003-06-17]

Glossar

Add-In	Optionales Zusatzmodul einer Software, mit dem die Funktionalität erweitert wird
Aktion	Elementare Prozess- Aktivität ohne weitere Untergliederung
Aktivität	UML-Element für einen Schritt in einem Prozessablauf
Anwendungsfall	UML-Element für die Beschreibung einer Funktion oder Aufgabe
ARIS	<i>Architektur Integrierter Informationssysteme</i> - Rahmenkonzept für die ganzheitliche Beschreibung computergestützter Informationssysteme in allen Entwicklungsphasen
CAD	<i>Computer-Aided Design</i> - Computergestützte Planung und Spezifikation von Produkten als Teil des Fertigungsprozesses
CSS	<i>Cascading Style Sheets</i> - Layout-Sprache zur Formatierung von Inhalten in XML-konformen Formaten
DESY	<i>Deutsches Elektronen Synchrotron</i> - Hamburger Forschungseinrichtung für Elementarteilchenphysik, Synchrotronstrahlung und Teilchen-Beschleuniger
DTD	<i>Document Type Definition</i> - Format zur Definition der Inhaltsstruktur von XML-Dokumenten
EDMS	<i>Engineering Data Management System</i> - Multi-user Datenbank für die Integration, inhaltliche Erschließung und zentralen Zugriff unterschiedlicher Dokumentformate im Rahmen der Fertigung
eEPK	<i>erweiterte Ereignisgesteuerte Prozesskette</i> - Methode zur Modellierung von Prozessabläufen; wird von ARIS verwendet
eERM	<i>erweitertes Entity-Relationship-Modell</i> - Methode zur Modellierung von Datenstrukturen
Help-Desk-System	Informationssystem zur Verwaltung von Vorgängen bei einer Service-Einrichtung mit direktem und häufigem Kundenkontakt
HTML	<i>Hypertext Markup Language</i> - Format zur Auszeichnung von Inhalten, die über IT-Netzwerke (Internet, Intranet) verbreitet und mit Hilfe von Web-Browsern gelesen werden
IPP	<i>Informationsmanagement, Prozesse, Projekte</i> - Organisationseinheit des DESY
Klasse	beinhaltet Objekte mit identischen Merkmalen und Verhalten; UML-Element
MOF	<i>Meta-Objekt Facility</i> - OMG-Standard für die Beschreibung von Metamodellen
NetInstall	Anwendung, mit der in einem Intranet Installationsdateien für Softwaresysteme zentral zur Verfügung gestellt und die lokalen Installationen verwaltet werden.

OLE	<i>Object Linking and Embedding</i> - Verfahren zur Integration verschiedener Dateiformate bei der Dokumenterstellung; betrifft vor allem Text-, Tabellen- und Grafikformate
OMG	<i>Object Management Group</i> - Internationales Konsortium von IT-Unternehmen, das Standards für die objektorientierte Softwareentwicklung erarbeitet
Package	UML-Element zur Zusammenfassung und Gliederung von Modellelementen
PDF	<i>Portable Document Format</i> - Dateiformat, mit dem Dokumente systemübergreifend publiziert werden können, wobei Layout-Informationen erhalten bleiben und die Bildschirmdarstellung der Druckfassung entspricht.
Rational Rose	Software für die visuelle Modellierung von Informationssystemen
Release	Produktversion einer Software
Repository	Strukturiertes Datenarchiv innerhalb einer Software oder Datenmanagementsystems
Request Tracker	Intranet-basierte Datenbank-Anwendung, mit der die Aufnahme, Weiterleitung und Bearbeitung von Anfragen verwaltet und dokumentiert wird
Schwimmbahn	Vertikales Gliederungselement in einem UML-Aktivitätsdiagramm
Service-Level-Agreements	Vereinbarungen über Leistungsstandards von Dienstleistungen
SVG	<i>Scalable Vector Graphics</i> – XML-Format zur Codierung von Grafiken in Form von Vektorkoordinaten
SVG-Parser	Programm zur Darstellung von Grafiken, die im Format SVG codiert sind.
SWOT-Analyse	Methode zur umfassenden Bewertung und zur Bestimmung von Entwicklungsperspektiven eines Produktes oder einer Organisation
Tagged Value	Eigenschaftsfeld eines UML-Elements, das nicht im Standard enthalten ist, sondern als freie Erweiterung hinzugefügt wurde
Toolbox	Anwendergerechte Aufbereitung eines Bündels von Editierungsfunktionen einer Software
UML	<i>Unified Modeling Language</i> - Visuelle Modellierungssprache für die Softwareentwicklung, Standard der OMG
URI	<i>Universal Resource Identifier</i> - Zeichenkette, die eine Ressource im Internet eindeutig definiert; benennt nacheinander Übertragungsprotokoll, Server, und Pfad zum Dokument, z.B. http://www.w3.org/TR/REC-html40/intro/intro.html
VKD	<i>Vorgangskettendiagramm</i> - ARIS-Methode, die Geschäftsprozesse aus verschiedenen Sichten global beschreibt.

WKD	<i>Wertschöpfungskettendiagramm</i> - ARIS-Methode, die den Ablauf betrieblicher Funktionen auf hohem Abstraktionsniveau beschreibt
Workflow	Spezifizierte und detaillierte Abfolge von Einzelschritten bei der Ausführung eines Prozesses; definiert u.a. die Kommunikation beteiligter Objekte und Akteure
XMI	<i>XML Metadata Interchange</i> - OMG-Standard auf der Basis von XML für die Beschreibung von Modellen und Metadaten
XML	<i>eXtensible Markup Language</i> - Auszeichnungssprache zur Strukturierung von Inhalten
XPath	<i>XML Path Language</i> - Teilsprache von XSL zur Adressierung von Inhalten in XML-Dateien
XSL	<i>XML Stylesheet Language</i> - Sprache zur Manipulation von XML-Daten, bestehend aus XSL-FO, XSLT und XPath
XSL-FO	<i>XML Stylesheet Language - Formatting Objects</i> - Teilsprache von XSL für die Gestaltung eines aus XML-Daten generierten Dokuments
XSLT	<i>XML Stylesheet Language - Transformation</i> - Teilsprache von XSL für die Transformation und Selektion von XML-Daten

A. Übersicht der vorgeschlagenen Modellierungskonventionen

Anwendungsbereich: Geschäftsprozessmodellierung
Modellierungssprache: UML 1.5
Modellierungswerkzeug: *Rational Rose 2000 Enterprise Edition*

(Konventionen in Fettdruck sind für die Darstellung im Publikationssystem obligatorisch)

Aufbau und Gliederung eines Modells

- i. **Das Grundgerüst eines Prozessmodells besteht aus einer hierarchischen Struktur von Aufgaben, die durch Anwendungsfälle repräsentiert werden.**
- ii. **Das Modell selbst steht an der Spitze der Hierarchie und wird durch einen Anwendungsfall des Stereotyps „business use case“ repräsentiert (Ebene 0).**
- iii. **Die Prozesse werden durch jeweils einen Anwendungsfall des Stereotyps „business use case“ repräsentiert (Ebene 1).**
- iv. **Aufgaben, die Teil eines Prozesses oder einer anderen Aufgabe sind, werden durch einfache Anwendungsfälle (ohne Stereotyp) dargestellt (Ebenen 2-N).**
- v. **Modell, Prozesse und Aufgaben werden durch Anwendungsfall-, Aktivitäts-, Sequenz- und Klassendiagramme näher beschrieben, die dem jeweiligen Anwendungsfall zugeordnet werden.**
- vi. Für jeden Prozess wird ein gleichnamiges Package in der „Use Case View“ angelegt. Die Anwendungsfälle (Prozess incl. Aufgaben und Teilaufgaben) eines Prozesses werden in das entsprechende Package gelegt. Der Anwendungsfall der Ebene 0 wird direkt in der „Use Case View“ angelegt.

Funktionssicht

Darstellung der Aufgabenhierarchie

- vii. **Die Teilaufgabenstruktur einer Aufgabe wird durch ein Anwendungsfalldiagramm dargestellt . Dazu wird dem Anwendungsfall einer Aufgabe mit Teilaufgaben genau ein Anwendungsfalldiagramm zugeordnet, das die Relationen zwischen Aufgabe (Ebene N) und deren Teilaufgaben (Ebene N+1) zeigt.**
- viii. Zwischen Anwendungsfällen werden nur die Relationen „verwendet“ oder „erweitert“ benutzt. Ein Anwendungsfall „verwendet“ andere Anwendungsfälle (bei obligatorischen Teilaufgaben) oder wird durch diese „erweitert“ (bei optionalen Teilaufgaben).

Mehrfachverwendung von Aufgaben in verschiedenen Kontexten

- ix. **Eine Aufgabe wird einem Prozess primär zugeordnet. Die Zuordnung wird in den Rängen der Anwendungsfälle definiert (s. Konvention xl).**
- x. Aufgaben werden in geeigneten Fällen auch in anderen als dem primär zugeordneten Prozess verwendet. Im Zweifelsfall wird ein neuer Anwendungsfall definiert.

Organisationssicht

Darstellung der Akteure

- xi. **Akteure der Prozesse werden durch Klassen des Stereotyps „Actor“ repräsentiert und in einem Package in der Logical View abgelegt.**
- xii. Für die Darstellung aller Akteure eines Modells einschließlich ihrer Beziehungen untereinander wird ein eigenständiges Klassendiagramm erstellt, das dem Anwendungsfall der Ebene 0 zugeordnet wird.
- xiii. Diejenigen Akteure, die für eine dargestellte Aufgabe eine wesentliche Durchführungsverantwortung wahrnehmen, werden auf den Anwendungsfalldiagrammen über ungerichtete Assoziationen mit den jeweiligen Anwendungsfällen verbunden.

Darstellung von Tätigkeitsprofilen

- xiv. Akteure erhalten keine eigenen Methoden, sondern die jeweiligen Klassen erben Methoden aus separaten Kenntnisklassen, die Cluster von Tätigkeiten repräsentieren.
- xv. Für ein Tätigkeitsprofil wesentliche Methoden werden in den Kenntnisklassen auch dann angelegt, wenn sie in keinem Szenario aufgerufen werden.

Objektsicht

Darstellung von Objekten

- xvi. **Alle Objekte der Prozesse werden durch Klassen repräsentiert und in der Logical View angelegt.**
- xvii. Klassen werden mit Hilfe von Packages in der Logical View nach formalen Gemeinsamkeiten unabhängig von den Prozessen grob gegliedert.
- xviii. Klassen, die innerhalb der Packages ein inhaltliches Cluster bilden, erhalten einen Stereotypen, der das Cluster anschaulich und kurz bezeichnet.
- xix. Die Beziehungen der Objekte untereinander sowie zwischen Objekten und Akteuren werden auf Klassendiagrammen dargestellt.
- xx. Klassendiagramme werden dem Anwendungsfall zugeordnet, der den engsten noch passenden Kontext für die abgebildete Klassenstruktur darstellt. Im Zweifelsfall werden sie der nächst höheren Ebene zugeordnet.

Modellierung von Systemoberflächen

- xxi. Die Objektstruktur von Informationssystemen mit prozessübergreifender Bedeutung wird in einem Klassendiagramm dargestellt, das dem Anwendungsfall der Ebene 0 zugeordnet ist.
- xxii. Jedes verwendete Dialogfenster wird durch eine eigene Klasse repräsentiert.
- xxiii. Die Anwenderfunktionen in den Dialogfenstern werden durch Methoden der jeweiligen Klassen repräsentiert (s Konventionen xxxiv-xxxvi).

Prozesssicht: Geschäftsprozessebene

- xxiv. Die Ablaufstruktur einer Aufgabe mit definierten Teilaufgaben wird durch ein Aktivitätsdiagramm dargestellt.
- xxv. Die Aktivitäten entsprechen jeweils in Benennung und Inhalt genau den Teilaufgaben der Aufgabenstruktur der zu beschreibenden Aufgabe.
- xxvi. Akteure in Aktivitätsdiagrammen werden durch Schwimmbahnen repräsentiert, die aus „Actor“- Klassen abgeleitet werden.
- xxvii. Objekte in Aktivitätsdiagrammen werden aus Klassen abgeleitet.

Prozesssicht: Workflow-Ebene

Darstellung von Handlungsanweisungen

- xxviii. Die Szenarien einer Ablaufstruktur und die Kommunikation beteiligter Objekte und Akteure bei der Ausführung einer Aufgabe werden durch Sequenzdiagramme dargestellt.
- xxix. Akteure und Objekte auf Sequenzdiagrammen werden aus Klassen abgeleitet.

Formale Definition der Nachrichten

- xxx. Eine Nachricht eines Szenarios ruft eine Methode des Nachrichtenempfängers (Objekt oder Akteur) auf, die in der jeweiligen Klasse definiert ist.
- xxxi. Daten, die in einer Nachricht übermittelt werden, werden durch Argumente der jeweiligen Methode dargestellt.

Inhalt und Gliederung der Nachrichten

- xxxii. Eine durch eine Nachricht aufgerufene Methode entspricht - entweder nach Inhalt und Benennung einer Aktivität einer Ablaufstruktur („Aktivitäts-Methode“); - oder sie definiert eine Aktion innerhalb einer Aktivität („Aktions-Methode“). Sie erhält analog einen der Stereotypen „Aktivität“ oder „Aktion“.
- xxxiii. Ein Szenario besteht entweder aus - einer Folge von Nachrichten mit Aktivitäts-Methode oder - einer Nachricht mit Aktivitäts-Methode, der Nachrichten mit zugehörigen Aktions-Methoden folgen oder - aus einer Kombination der beiden ersten Varianten

Bedienung von Informationssystemen

- xxxiv. Die Aktivierung eines Buttons oder Hyperlinks auf einer Systemoberfläche wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „click: [Button- oder Hyperlink-Name]()“ aufruft.
- xxxv. Die Aktivierung einer Menüfunktion wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „click: [menüpfad]/[menüfunktion]()“ aufruft.
- xxxvi. Die Änderung eines Eingabe- oder Auswahlfeldes auf einer Systemoberfläche wird durch eine Nachricht repräsentiert, die eine Methode nach dem Muster „edit: [Feldname]=[Eingabe]()“ aufruft.

Weitere Anforderungen der Prozessdokumentation

Beschreibungstexte

- xxxvii. Die Präambel des Modells wird im Dokumentationsfeld der „Use Case View“ niedergelegt.**
- xxxviii. Jede Aufgabe erhält obligatorisch eine Beschreibung im Dokumentationsfeld des jeweiligen Anwendungsfalles.**
- xxxix. Objekte, Akteure und Methoden erhalten eine Beschreibung im Dokumentationsfeld der jeweiligen Klassen und Methoden. Davon kann abgesehen werden, wenn die Bedeutung offensichtlich oder aus dem Kontext leicht zu erschließen ist.**

Steuerung der Print-Dokumentation

- xl. Jede Aufgabe erhält eine dreiteilige numerische Kennung nach dem Muster xxyyzz im Feld „Rank“ des jeweiligen Anwendungsfalles. Der erste Teil (xx) kennzeichnet den Geschäftsprozess, dem die Aufgabe primär zugeordnet ist, der zweite Teil (yy) die Ebene in der Hierarchie und der dritte Teil (zz) benennt den Rang der Bedeutung innerhalb einer Ebene eines Geschäftsprozesses. Der Anwendungsfall der Ebene 0 erhält die Kennung „000000“.**

Einbindung von Dokumenten

- xli. Dateien und Intranetquellen mit Hilfs- und Arbeitsmitteln werden immer entweder Szenarien, Aufgaben oder Objekten zugeordnet. Entsprechende Verweise werden daher den jeweiligen Sequenzdiagrammen, Anwendungsfällen oder Klassen angehängt.**

B. Codierung des XML-Schemas

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xsd:element name="prozessmodell" type="modell1"/>
  <xsd:complexType name="modell1">
    <xsd:sequence>
      <xsd:element name="ModellID" type="xsd:string"/>
      <xsd:element name="ModellName" type="xsd:string"/>
      <xsd:element name="ModellDoku" type="xsd:string"/>
      <xsd:element name="UCViewID" type="xsd:string"/>
      <xsd:element name="AwFall" type="AFGruppe" maxOccurs="unbounded"/>
      <xsd:element name="Klasse" type="KlassenGruppe" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Diagrammgruppe" mixed="true">
    <xsd:sequence>
      <xsd:element name="Diagramm" type="Diagramm" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Diagramm">
    <xsd:sequence>
      <xsd:element name="DiaID" type="xsd:string"/>
      <xsd:element name="DiaName" type="xsd:string"/>
      <xsd:element name="DiaDoku" type="xsd:string"/>
      <xsd:element name="DiaExt" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
      <xsd:element name="DiaAreas" type="xsd:string"/>
      <xsd:element name="Dialtem" type="Item" maxOccurs="unbounded"/>
      <xsd:element name="Objekt" type="Objekt" minOccurs="0"
maxOccurs="unbounded"/>
      <xsd:element name="Nachricht" type="Nachricht" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="Item" mixed="true">
    <xsd:sequence>
      <xsd:element name="ItemID" type="xsd:string" minOccurs="0"/>
      <xsd:element name="ItemName" type="xsd:string" minOccurs="0"/>
      <xsd:element name="ItemStereo" type="xsd:string" minOccurs="0"/>
      <xsd:element name="Verweis" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="AFGruppe">
    <xsd:sequence>
      <xsd:element name="AFID" type="xsd:string"/>
      <xsd:element name="AFName" type="xsd:string"/>
      <xsd:element name="AFStereo" type="xsd:string"/>
      <xsd:element name="AFDoku" type="xsd:string"/>
      <xsd:element name="AFRang" type="xsd:number"/>
      <xsd:element name="AFPackPfad" type="xsd:string"/>
      <xsd:element name="AFExt" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
      <xsd:element name="AFDiaAF" type="Diagrammgruppe"/>
      <xsd:element name="AFDiaAkt" type="Diagrammgruppe"/>
    </xsd:sequence>
  </xsd:complexType>

```

```
<xsd:element name="AFDiaSeq" type="Diagrammgruppe"/>
  <xsd:element name="AFDiaKlass" type="Diagrammgruppe"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="KlassenGruppe">
  <xsd:sequence>
    <xsd:element name="KlassenID" type="xsd:string"/>
    <xsd:element name="KlassenName" type="xsd:string"/>
    <xsd:element name="KlassenStereo" type="xsd:string" minOccurs="0"/>
    <xsd:element name="KlassenDoku" type="xsd:string"/>
    <xsd:element name="KlassenExt" type="xsd:anyURI" minOccurs="0"
maxOccurs="unbounded"/>
    <xsd:element name="Operation" type="OpGruppe" minOccurs="0"
maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OpGruppe">
  <xsd:sequence>
    <xsd:element name="OPID" type="xsd:string"/>
    <xsd:element name="OPName" type="xsd:string"/>
    <xsd:element name="OPDoku" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Objekt">
  <xsd:sequence>
    <xsd:element name="ObjektID" type="xsd:string"/>
    <xsd:element name="KlassenID" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Nachricht">
  <xsd:sequence>
    <xsd:element name="NachrichtID" type="xsd:string"/>
    <xsd:element name="OPID" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

C. Grafische Darstellung des XML-Schemas

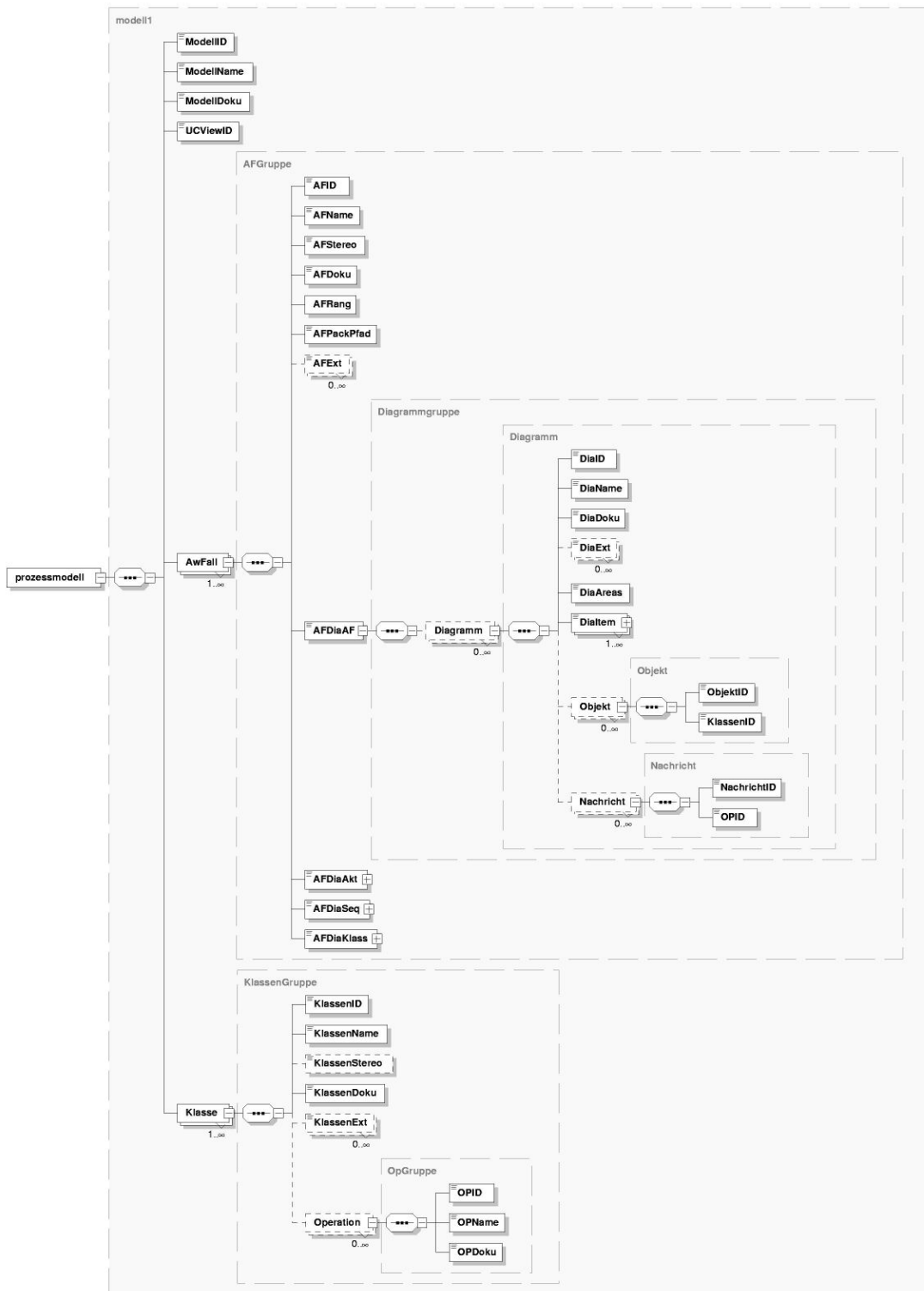


Abbildung 76: Strukturbaum des proprietären XML-Schemas

Erklärung zur Diplomarbeit:

Ich versichere, dass ich die Arbeit selbständig, ohne fremde Hilfe verfasst habe.

Bei der Abfassung der Arbeit sind nur die angegebenen Quellen und Hilfsmittel benutzt worden.

Ich bin damit einverstanden, dass meine Arbeit veröffentlicht wird.

Hamburg, 28.07.03