

Flexibles Workflowmanagement für Entwurfsprozesse am Beispiel der Bebauungsplanung¹

Dr. Frank Maurer & Dipl.-Inform. Gerhard Pews²
Universität Kaiserslautern
AG Expertensysteme Prof. Richter
Postfach 3049
67653 Kaiserslautern

Zusammenfassung

In diesem Beitrag wird ein Ansatz vorgestellt, mit dem Entwurfsprozesse geplant, koordiniert und durchgeführt werden können. In der Projektplanung wird zunächst ein explizites Modell des Entwurfsprozesses erstellt. Dieses wird von einem Workflowmanagementsystem als Basis für die Projektdurchführung benutzt. Projektplanung und -durchführung können miteinander verzahnt werden, d. h. während der Durchführung kann weiter-, um- und feingeplant werden. Während der Projektabwicklung werden kausale Abhängigkeiten zwischen (Teil-) Ergebnissen (wie z. B. Anforderungen, Vorentwürfen und Zeichnungen) fein-granular erfaßt, repräsentiert und verwaltet. Diese Abhängigkeiten werden im wesentlichen automatisch aus dem Prozeßmodell abgeleitet. Durch die Repräsentation der kausalen Abhängigkeiten wird die Verfolgbarkeit des Entwicklungsprozesses erhöht und dessen Ergebnisse dokumentiert. Basierend auf den repräsentierten Abhängigkeiten kann auf Änderungen gezielt reagiert werden, wodurch die Entwicklungskosten sinken. Die entwickelten Techniken werden am Beispiel der Bebauungsplanung erläutert.

Abstract

In this paper we describe an approach for planning, coordination and processing of design processes. In the project-planning stage, an explicit model describing the design process is acquired and will be used as the basis for process execution by a workflow-management-system. During process execution this plan can be modified in order to make it more detailed, to adapt

¹ Die hier beschriebene Arbeit wurde teilweise von der VW-Stiftung unter Vertrag I/69 374 gefördert im Rahmen des interdisziplinären Projektes „Intelligenter Bebauungsplan“. Die Projektpartner sind die Arbeitsgruppen Prof. Richter (Informatik), Prof. Stich (ARUBI) und Prof. Streich (ARUBI) an der Universität Kaiserslautern.

² In alphabetischer Reihenfolge.

to the current situation or to carry on planning from a point where product data is required for further decisions. The workflow-management-system automatically generates causal dependencies between (partial) results from the process model on a detailed level. These dependencies are represented and managed in order to make the whole design process more traceable and to document the results. If changes take place, these dependencies are used to react selectively and to calculate and manage their consequences, which reduces development costs. We illustrate these techniques using one of our application domains, urban land-use planning.

1. Einleitung

Entwurfsprozesse bestehen zum einen aus graphisch orientierten Arbeitsschritten. Zum anderen werden auch nicht-graphische Informationen erzeugt. Um den gesamten Entwurfsprozeß durchzuführen, zu koordinieren, zu verstehen und zu optimieren ist es notwendig, den gesamten Arbeitsablauf und alle darin erzeugten Informationen zu beschreiben und zu verwalten. Workflowmanagement-Ansätze unterstützen diese Gesamtsicht auf Arbeitsabläufe bisher für Routinetätigkeiten und nicht für kreative Prozesse. Langfristiges Ziel unserer Arbeit ist nun die Entwicklung von Methoden, Techniken und Systemen zur Unterstützung von komplexen kooperativen Entwurfs- und Konstruktionsprozessen. Charakterisiert werden diese Prozesse durch folgende

Eigenschaften:

- Mehrere Bearbeiter arbeiten zeitlich und/oder räumlich verteilt an einer gemeinsamen Aufgabe. Der daraus resultierende hohe Koordinierungs-, Entwicklungs- und Wartungsaufwand verursacht einen Großteil der Kosten.
- Entwicklungsprojekte ziehen sich oft über einen längeren Zeitraum hinweg und sind geprägt durch sich häufig ändernde Anforderungen an das zu erstellende Produkt. Durch eine gezielte Reaktion auf Anforderungsänderungen, die die Verfolgbarkeit der Entwicklungsprozesse voraussetzt, lassen sich somit die Kosten drastisch reduzieren.
- Im Verlauf des Entwicklungsprozesses werden von den beteiligten Personen eine große Zahl von Einzelentscheidungen getroffen, die sich gegenseitig beeinflussen und aufeinander aufbauen.
- Im Verlauf des Entwurfsprozesses müssen die Bearbeiter auf komplexe und umfangreiche Informationen („Wissen“) wie z. B. Normen, Richtlinien, Lehrbücher zurückgreifen.

Aus diesen charakteristischen Eigenschaften von Entwurfsprozessen läßt der Ausgangspunkt unserer Arbeiten ableiten, die

Anforderungen an Planungs- und Abwicklungswerkzeuge:

- Die räumliche und zeitliche Verteilung der Arbeitsprozesse erfordert, daß das Werkzeug Entscheidungen dokumentiert, damit andere oder spätere Benutzer sie nachvollziehen können.

- Das Werkzeug soll als „intelligentes Gedächtnis“ seiner Benutzer dienen und verwalten, auf welcher Basis Einzelentscheidungen beruhen.
- Das Werkzeug soll nach Änderungen automatisch die davon betroffenen Benutzer informieren.

Um komplexe Arbeitsabläufe und Entscheidungsprozesse plan-, steuer- und kontrollierbar zu machen, wurden verschiedene Techniken entwickelt. Den Ausgangspunkt der hier vorgestellten Arbeit bildeten:

- *Workflowmanagement* (GeH95), (Jab95) unterstützt die Durchführung von Routinetätigkeiten (repetitive Prozesse), wie z. B. die Bearbeitung eines Schadensfalls in einer Versicherung oder das Erteilen einer Baugenehmigung in einer Baubehörde. Grundlage eines Workflowmanagementsystems ist die Modellierung der Geschäftsprozesse. Die Arbeitsabläufe müssen in der Regel definiert werden, bevor mit ihrer Durchführung begonnen werden kann. Entwurfsprozesse sind allerdings gekennzeichnet durch die Verzahnung von Planungs- und Ausführungsschritten, da oft die weitere Projektplanung von den Ergebnissen der bereits durchgeführten Arbeitsschritte abhängig ist.
- *Hypertext/Hypermedia-Techniken* werden eingesetzt, um komplexes Wissen übersichtlich zu vermitteln. Dokumentationsrichtlinien, Normen, Gesetze etc., die in Form von HTML-Dokumenten über das World-Wide-Web (WWW) zugreifbar sind, können zentral für ein ganzes Unternehmen verwaltet werden. Dadurch läßt sich der Aufwand für Aktualisierungen drastisch verringern.
- Der Begriff „*Design Rationale*“ (AAAI92), (BSH94), (SDR91) faßt Arbeiten zusammen, die sich mit dem Erfassen und der Verwaltung von fein-granularen Entwurfsentscheidungen und den zwischen ihnen bestehenden Abhängigkeiten beschäftigen. Ein Problem der bisherigen Ansätze ist, daß Begründungen interaktiv vom Benutzer eingegeben werden müssen. Dies verursacht ihm einen hohen zusätzlichen Aufwand und reduziert die Akzeptanz der entwickelten Systeme.

In diesem Beitrag werden Methoden und Techniken beschrieben, die die Planung und Abwicklung komplexer Entwurfsprozesse unterstützen. Aus der Zerlegung von Entwicklungsaufgaben und aus dem Informationsfluß fein-granularer Projektpläne von Entwicklungsprozessen werden automatisch kausale Abhängigkeiten zwischen Entwurfsentscheidungen abgeleitet. Diese Abhängigkeiten werden benutzt um auf Änderungen zu reagieren, indem die davon betroffenen Benutzer darüber in Kenntnis gesetzt werden. Diese können dann die getroffenen Entscheidungen gezielt überdenken.

Das von uns entwickelte System verwaltet den aktuellen Stand des Entwurfsprozesses und alle erzeugten Informationen. Die getroffenen Entscheidungen und die darin eingehenden Voraussetzungen werden automatisch im Sinne eines umfassenden Qualitätsmanagements in Konstruktionsprozessen dokumentiert. Dadurch werden die erzielten Resultate nachvollziehbar.

Der folgende Abschnitt gibt einen Überblick über unseren CoMo-Kit-Ansatz. Anschließend stellt Abschnitt 3 die Modellierungs-/Projektplanungswerkzeuge von CoMo-Kit vor. Der vierte Abschnitt beschreibt den Formalismus zur Verwaltung kausaler Abhängigkeiten und wie diese akquiriert werden. Er enthält den methodischen Kern des Papiers. Den entwickelten

Workflowmanager, der Abhängigkeiten zwischen Entscheidungen verwaltet und so die Abwicklung komplexer Entwurfsprozesse unterstützt, stellt Abschnitt 5 vor. Unsere Anwendungsdomäne, die Bebauungsplanung als Teil des Stadtplanungsprozesses, wird im sechsten Abschnitt vorgestellt. Der letzte Abschnitt faßt unsere Ergebnisse zusammen und zeigt noch offene Probleme auf.

2. CoMo-Kit Übersicht

Im Rahmen des CoMo-Kit Projektes der Universität Kaiserslautern werden Techniken, Methoden und Systeme entwickelt, die zum einen die Projektplanung und zum anderen die Projektdurchführung komplexer verteilter Entwurfsprozesse unterstützen (Mau93), (DMP95). Für ein (neues) Projekt wird in einem ersten Schritt ein Modell erstellt: der initiale Projektplan. Dieser umfaßt im wesentlichen die zu erledigenden Aufgaben, die Struktur der zu erzeugenden Informationen und die Beschreibung der beteiligten Mitarbeiter. Für den Bebauungsplanungsprozeß wurde von uns ein universelles Modell erstellt. Dieses wird im weiteren Verlauf eines konkreten Projektes weiter verfeinert und dem aktuellen Projektverlauf angepaßt, d.h. unser Ansatz verzahnt Planung und Ausführung und bietet dafür Computerunterstützung an. *Wir gehen davon aus, daß komplexe Entwurfsprozesse immer erst während der Abwicklung detailliert geplant werden können.* Da konventionelle Workflowsysteme davon ausgehen, daß der Arbeitsablauf vollständig definiert ist, bevor er startet, können diese nur schwer zur Unterstützung von Konstruktionsprozessen herangezogen werden.

Der (aktuelle) Projektplan wird von unserem Workflow-Management-Server, dem CoMo-Kit Scheduler, eingesetzt, um die Projektabwicklung zu unterstützen. Verschiedene Clients können benutzt werden, um Teilaufgaben zu bearbeiten. So wurde z. B. ein Client entwickelt, der den zeichnerischen Teil des Bebauungsplanungsprozesses unterstützt.

Die entwickelten Techniken werden im Moment in zwei Anwendungsgebieten eingesetzt: in der Bebauungsplanung (MPe95) und im Software Engineering.

Abbildung 1 zeigt die Gesamtarchitektur der CoMo-Kit Systems. Im groben besteht es aus drei Teilen:

- Der *CoMo-Kit Modeler* ermöglicht die Erstellung von Projektplänen.
- Der *CoMo-Kit Scheduler* unterstützt die Abwicklung von Projektplänen, wobei durch den Rückgriff auf den Modeler die Pläne jederzeit verändert werden können. Der Server verwaltet den aktuellen Projektzustand und alle im Projektverlauf erzeugten Informationen. Die Clients unterstützen die Bearbeitung einzelner Teilaufgaben.
- Der *CoMo-Kit Information Assistent* erlaubt den Zugriff auf den aktuellen Projektzustand und die Begründungen von Entwurfsentscheidungen. Im Rahmen unseres Projektes „Intelligenter Bebauungsplan“ bildet er die Basis eines Bürgerinformationssystems. Er wird im folgenden nicht näher beschrieben.

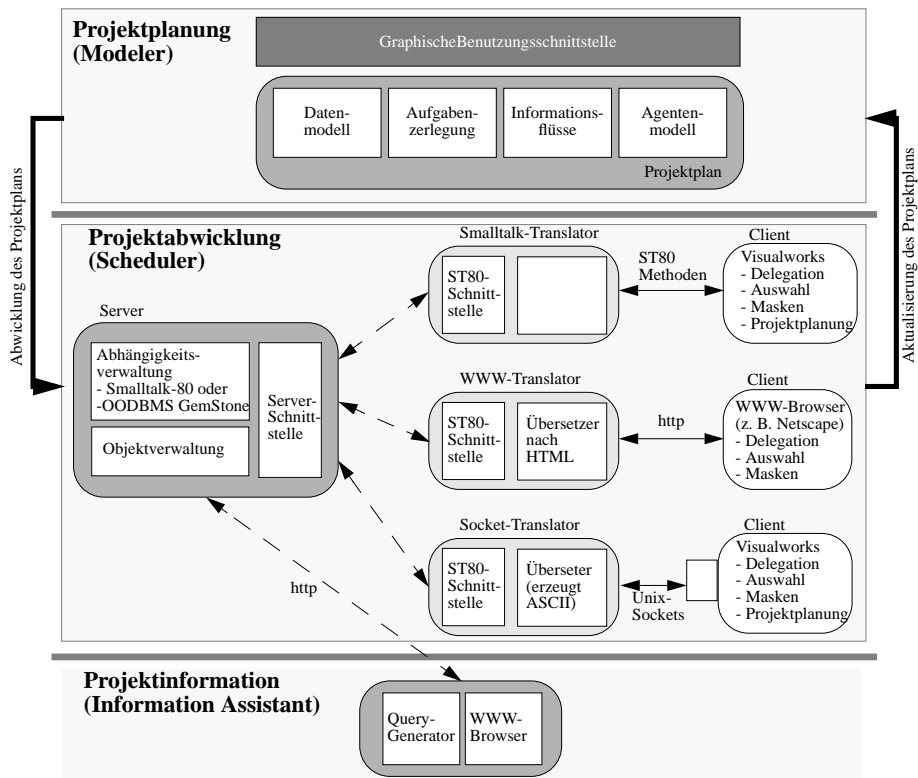


Abbildung 1: Die Architektur des CoMo-Kit Systems

3. Modellierung/Projektplanung mit CoMo-Kit

Um kooperative Entwurfsprozesse zu modellieren, benutzt unsere Methodik vier Grundbegriffe: Aufgabe (task), Methode (method), Konzept (concept) und Agent (agent). Im folgenden werden wir diese, soweit es zum Verständnis dieses Beitrags erforderlich ist, definieren.

Aufgaben. Eine Aufgabe beschreibt ein zu erreichendes Ziel. Ihr sind Eingaben und Ausgaben zugeordnet. Bei der Bearbeitung einer Aufgabe wird eine Entscheidung getroffen, die sich in der Zuweisung von Werten an die Ausgaben manifestiert. *Unser Ansatz geht davon aus, daß zwischen den Eingaben und den Ausgaben ein kausaler Zusammenhang existiert.* Dem liegt

die Annahme zugrunde, daß bei der Erstellung des Projektplans einer Aufgabe nur Eingaben zugeordnet werden, die für die Bearbeitung der Aufgabe notwendig sind.

Methoden. Um eine Aufgabe zu bearbeiten wird eine Methode angewandt. Für jede Aufgabe kann es eine Menge alternativer Methoden geben. Methoden werden von Agenten ausgeführt. Wir unterscheiden zwischen atomaren und komplexen (zusammengesetzten) Methoden. Atomare Methoden weisen Variablen aktuelle Werte zu. Eine komplexe Methode zerlegt eine Aufgabe in kleinere Teilaufgaben. Sie wird als Informationsflußgraph beschrieben. Ein Informationsflußgraph besteht aus Knoten, die (Unter-)Aufgaben und Variable definieren, und aus Kanten, die den Aufgaben ihre Ein- bzw. Ausgaben zuordnen. Jeder Variable ist ihr Typ zugeordnet, d.h. die Konzeptklasse (s. u.), die zur Laufzeit instanziiert und deren Instanz dann an die Variable gebunden wird. Jede Unteraufgabe kann mit Hilfe von Methoden weiter zerlegt werden. Die für die Lösung einer Aufgabe sinnvolle Methode wird erst während der Projektdurchführung bestimmt. Abbildung 4 zeigt einen Informationsfluß unserer Anwendungsdomäne, der Bebauungsplanung.

Konzepte. Im Verlauf eines Projektes zu erzeugende Produkte werden mit einem objekt-zentrierten Ansatz modelliert, wobei zwischen Klassen und Instanzen unterschieden wird. Die Klassen sind in einer Vererbungshierarchie angeordnet, sie definieren die Struktur der Instanzen durch eine Menge von Slots. Jedem Slot wird ein Typ und eine Kardinalität zugeordnet. Der Typ eines Slots kann dabei entweder ein Basistyp wie REAL, STRING, etc. oder wiederum ein anderes Konzept sein. Dadurch ergeben sich ineinander geschachtelte Objektstrukturen. In der Visualisierung der Informationsflüsse können die Ein-/Ausgaben dadurch in verschiedenen Detaillierungsgraden angezeigt werden.

Agenten. Aufgaben werden von Agenten bearbeitet. Im konzeptuellen Modell wird für jede Aufgabe definiert, welche Agenten fähig sind, sie zu bearbeiten. Wir unterscheiden dabei zwischen zwei verschiedenen Arten von Agenten: Menschen und Rechner. In Abhängigkeit von den zugeordneten Agenten werden Methoden in einer natürlichen oder formalen (operationalen) Sprache beschrieben.

Der Plan eines Entwurfsprozesses umfaßt nun zu bearbeitende Aufgaben, deren Zerlegungen und den Informationsfluß zwischen Aufgaben.³ Aus diesen Plänen lassen sich kausale Abhängigkeiten ableiten, die wir im folgenden Abschnitt vorstellen.

4. Formale Repräsentation kausaler Abhängigkeiten

Einen Schwerpunkt unserer Arbeit bilden Methoden, die kausale Abhängigkeiten zwischen Entwurfsentscheidungen erfassen und verwalten. Dies ist notwendig, um die Nachvollziehbarkeit des Entwurfsprozesses gewährleisten zu können und um auf geänderte Anforderungen gezielt zu reagieren.

³ Auf die Zuordnung von Aufgaben zu Bearbeitern gehen wir nicht ein.

Ein wesentlicher Teil dieser Abhängigkeiten und deren Begründungen wird automatisch aus dem Projektplan abgeleitet, so daß die Benutzer von der manuellen Eingabe von Rechtfertigungen entlastet werden.

Aus dem Projektplan leitet unser Ansatz automatisch zwei verschiedene Arten von Abhängigkeiten ab:

- Abhängigkeiten aufgrund des Informationsflusses und
- Abhängigkeiten aufgrund der Aufgabenzerlegung und der Objektzerlegung.

Im folgenden erläutern wir den unserem System zugrundeliegenden Formalismus zur Verwaltung von Abhängigkeiten.

Bei der Bearbeitung einer Aufgabe wird eine *Entscheidung* getroffen. Diese resultiert in der *Belegung von Variablen* oder in der *Zerlegung der Aufgabe in Teilaufgaben*. Begründungen sind Argumente für oder gegen eine Entscheidung bzw. die Gültigkeit einer Information. Entscheidungen können sich im Projektverlauf als falsch oder suboptimal herausstellen. Sie können dann zurückgezogen werden und werden dadurch ungültig.

Um die genannten Begriffe zu formalisieren, wurde eine Menge von Prädikaten definiert. Diese können *gültig* (TRUE) oder *ungültig* (FALSE) sein. Eine Begründung ist eine *logische Implikation* zwischen Prädikaten.

Wenn eine Entscheidung m getroffen wird, dann wird das zugeordnete Prädikat $decision(m)$ gültig.

Wenn es einen Grund gegen die Entscheidung für die Methode m gibt, dann wird das Prädikat $rejected-decision(m)$ gültig und die Entscheidung wird zurückgezogen. Dies wird automatisch erreicht durch in der folgenden Implikation ausgedrückte Abhängigkeit:

$$rejected-decision(m) \Rightarrow \neg decision(m) \quad (GL 1)$$

Das Prädikat $rejected-decision(m)$ kann selbst wieder formal begründet werden. Wenn die dabei benutzten Prädikate dann ungültig werden, wird automatisch auch die Entscheidung $decision(m)$ zurückgezogen.

Die beiden Prädikate dienen als Anknüpfungspunkte für kausale Abhängigkeiten. Wie diese automatisch aus dem Projektplan extrahiert werden können, wird im folgenden dargestellt.

4.1. Abhängigkeiten aufgrund des Informationsflusses

Bei der Bearbeitung einer Aufgabe werden vorhandene Informationen benutzt, um neue Informationen zu erzeugen. Der Projektplan beinhaltet Informationsflüsse, die diese Beziehung darstellen. *Wir gehen davon aus, daß die erzeugten Informationen kausal von den benutzten abhängig sind.* Das hat zur Folge, daß die Gültigkeit der erzeugten Informationen in Frage gestellt werden muß, wenn die benutzten ungültig werden. Einen Formalismus, der diesen Effekt erzeugt, stellen wir im folgenden an einem kleinen Beispiel dar.

Gegeben sei der in Abbildung 2 gezeigte Ausschnitt eines Informationsflusses. In Aufgabe T wird die in Eingabevariable I gespeicherte Information benutzt um die Belegung der Ausgabevariablen O_1 und O_2 zu erzeugen. Wird nun eine Methode m_j angewendet um die Aufgabe T zu lösen, dann werden dadurch die Variablen O_1 und O_2 an die Werte o_{1j} bzw. o_{2j} gebunden.

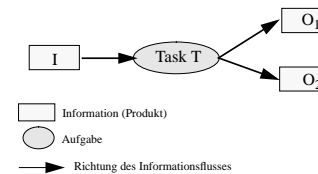


Abbildung 2: Informationsfluß zu einer

Formal wird dies allgemein mit der Formel aus Gleichung 2 repräsentiert. Dabei drückt die Gültigkeit des Prädikats $assignment(O_j=o_{1j})$ aus, daß die Variable O_j mit dem Wert o_{1j} belegt ist.

$$decision(m_j) \Rightarrow assignment(O_1=o_{1j}) \wedge assignment(O_2=o_{2j}) \wedge \dots \wedge assignment(O_v=o_{vj}) \quad (GL 2)$$

Verliert nun das Prädikat $decision(m_j)$ seine Gültigkeit, so werden automatisch auch die Belegungen der Ausgabevariablen ungültig. Unser Ziel war nun, die Gültigkeit der Ausgaben von den Eingaben abhängig zu machen. Formal läßt sich das mit Hilfe von Gleichung 3 in Zusammenarbeit mit Gleichung 1 erreichen.

$$\neg assignment(I_1=i_{1k}) \vee \dots \vee \neg assignment(I_n=i_{nk}) \Rightarrow rejected-decision(m_j) \quad (GL 3)$$

Die Gleichungen 1-3 werden von unserem System jeweils erzeugt, sobald eine Entscheidung für eine atomare Methode getroffen wird. Da die Ausgabevariablen einer Aufgabe, die Eingaben von anderen Aufgaben sind, entstehen im Laufe des Entwicklungsprozesses lange Ketten von Abhängigkeiten. Die Gleichungen repräsentieren somit ein allgemeines Schema zur Darstellung kausaler, auf dem Informationsfluß beruhender Abhängigkeiten.

4.2. Abhängigkeiten aufgrund der Aufgabenzerlegung

Zusätzlich zu den Abhängigkeiten aufgrund des Informationsflusses und der Objektzerlegung, verwaltet unser System noch die Gültigkeit von Aufgaben. Dadurch können nach einer Umplanung automatisch die Mitarbeiter informiert werden, die noch an einer nun mehr ungültigen Aufgabe arbeiten. Dies vereinfacht und (teil-) automatisiert die Projektkoordination.

Komplexe Methoden zerlegen Aufgaben in mehrere, kleinere Teilaufgaben. Wird während der Projektabwicklung von einem Mitarbeiter eine Entscheidung für eine Methode m_i getroffen, dann werden die Teilaufgaben $T_1 \dots T_n$ gültig und müssen im weiteren Projektverlauf bearbeitet

werden. Formal wird dieser Zusammenhang durch Gleichung 4 ausgedrückt. Das Prädikat $validSubtask(T_i)$ ist gültig, solange die Aufgabe T_i im Rahmen des Entwurfsprozesses bearbeitet werden muß.

$$\begin{aligned} decision(m_i) \Rightarrow \\ & validSubtask(T_1) \wedge \\ & validSubtask(T_2) \wedge \dots \\ & validSubtask(T_n) \end{aligned} \quad (GL\ 4)$$

Die Auswahl einer komplexen Methode wird von uns als Planungsentscheidung angesehen. Diese ist in der Regel nicht abhängig von den Belegungen der Eingabevariablen der Aufgabe. Die in Gleichung 3 dargestellte Beziehung zwischen Eingaben und dem Rückzug einer Entscheidung wird daher für komplexe Methoden nicht erzeugt.

Abhängigkeiten aufgrund der Objektstruktur

Wertzuzuweisungen werden durch einfache Assignments repräsentiert. Die Produkte sind aber objektzentriert beschrieben. Dadurch tritt das Problem auf, daß zwischen den Assignments noch Abhängigkeiten bestehen, die von der Objektstruktur herrühren. Wird z. B. ein Produkt erzeugt, das bestimmte Eigenschaften besitzt, werden die Assignments der Eigenschaften ungültig, sobald das Produkt selbst nicht mehr existiert (d. h. dessen Assignment ungültig wird). Da jedoch die Produkte durch die Bearbeitung der einzelnen Aufgaben (indem dort eine Entscheidung für eine Methode getroffen wird) sukzessive aufgebaut werden, ist diese Abhängigkeit bereits durch die im vorigen Abschnitt beschriebenen Gleichungen implizit repräsentiert.

Diese Abhängigkeiten aufgrund der Objektstruktur widersprechen nicht der Tatsache, daß man im Entwurfsprozeß Vorgaben machen kann, die erst später zu erstellende Produkte betreffen. Diese Vorgaben können natürlich, da sie selbstständige Objekte sind, ohne das Produkt existieren. Erst wenn sie in einem späteren Arbeitsschritt in das Produkt eingearbeitet werden und somit dann Teil des Produkts sind, wird die strukturbedingte Abhängigkeit relevant.

4.3. Grenzen der automatischen Ableitbarkeit kausaler Abhängigkeiten

Die in den Abschnitten 4.1 - 4.2 beschriebenen Abhängigkeiten werden von unserem System automatisch aus einem Projektplan abgeleitet. In der Praxis zeigt sich aber leider, das Projektpläne in der Regel nicht alle oder zuviele Abhängigkeiten enthalten. Diese Unzulänglichkeiten haben verschiedene Ursachen:

- Wird für das aktuelle Projekt ein Standardplan genommen (z. B. kann der Entwurfsprozeß in einem Qualitätsmanagementhandbuch nach DIN/ISO 9000ff vorgegeben werden), so ist damit zu rechnen, daß er auf das aktuelle Projekt nicht genau paßt oder zu grob ist.
- Große Projekt ziehen sich über einen längeren Zeitraum hin. Am Anfang kann dabei der gesamte Projektverlauf nur grob geplant werden. Für die ersten Schritte ist evtl. auch eine feinere Planung möglich. Die späteren Schritte sind allerdings erst detailliert planbar, wenn die Ergebnisse der ersten vorliegen.

Während man beim ersten Punkt argumentieren könnte, daß ein „hinreichend guter“ Standardplan die erwähnte Unzulänglichkeit nicht aufweisen würde, so ist der zweite Punkt eher prinzipieller Natur und kann *nie* durch eine Vorabplanung gelöst werden.

Es kann also vorkommen, daß eine kausale Abhängigkeit zwischen zwei Informationen nicht modelliert ist oder daß in eine Aufgabe eine Information eingeht, die zu ihrer Bearbeitung irrelevant ist. Daher sieht unser System vor, daß die Benutzer während der Projektabwicklung zusätzliche Begründungen (Design Rationales) für jede Entscheidung angeben bzw. überflüssige streichen können.

Formal werden zusätzliche Begründungen in das System aufgenommen, indem Gleichung 3 erweitert wird. Der Rückzug einer Entscheidung kann abhängig gemacht werden von

- der Gültigkeit bzw. Ungültigkeit anderer Entscheidungen und
- der Gültigkeit bzw. Ungültigkeit von Variablenbelegungen.

Gleichung 5 zeigt die Erweiterung der Gleichung 3. Natürlich müssen die Benutzer die Formel nicht direkt eingeben, sondern mit Hilfe einer graphischen Benutzungsschnittstelle.

$$\begin{aligned} & \neg assignment(I_i=i_{ik}) \vee \dots \vee \neg assignment(I_n=i_{nk}) \vee \\ & \neg decision(m_{alt1}) \vee \dots \vee \neg decision(m_{altM}) \\ & decision(m_{alt1}) \vee \dots \vee decision(m_{altM}) \\ & \neg assignment(X_I=x_{alt1}) \vee \dots \vee \neg assignment(X_M=x_{altM}) \\ & assignment(X'_I=x'_{alt1}) \vee \dots \vee assignment(X'_M=x'_{altM}) \\ & \Rightarrow rejected_decision(m_{neu}) \end{aligned} \quad (GL\ 5)$$

5. Workflowmanagement mit CoMo-Kit

CoMo-Kit verfügt neben dem Werkzeug zur Erstellung von Prozeßmodellen über ein Workflowmanagement-System, den CoMo-Kit-Scheduler. Der Scheduler operationalisiert die entwickelten Prozeßmodelle und erlaubt die interaktive Ausführung von Planungs- und Entwicklungsprozessen (DMP95), (MPa94). Er verwaltet den aktuellen Stand des Planungs- und Entwicklungsprozesses, die im Laufe des Lösungsprozesses erzeugten Informationen, die Abhängigkeiten zwischen Entwurfsentscheidungen und unterstützt den Rückzug von Entscheidungen.

Verwaltung des Entwurfsprozesses und der anfallenden Entscheidungen

In komplexen Entwurfsprozessen müssen eine Unmenge von Aufgaben und Entscheidungen verwaltet und koordiniert werden, die durch komplexe Informationsflüsse in Beziehung zueinander stehen. Es ist daher unumgänglich, die große Menge der sich ergebenden Abhängigkeiten effizient zu verwalten. Wir nutzen dabei die Eigenschaft von Entwurfsprozessen aus, daß die meisten Änderungen nur wenige Objekte betreffen und sich daher nur bei einem kleinen Teil der Assignment-Prädikate der logische Zustand ändert.

Als Basismodell dient uns das allgemeine Planungs- und Designmodell REDUX (Pet91). Der Scheduler erweitert die Funktionalität von REDUX um die Verwaltung von Informationsflüs-

sen und den sich daraus ergebenden Abhängigkeiten. Die Verwaltung der logischen Formeln übernimmt ein Truth Maintenance System (JTMS (Doy79)), das in der Lage ist, Änderungen der logischen Werte von Formeln lokal zu propagieren.

Verzahnung von Planung und Abwicklung

Ein Entwurfsprozeß ist auf detaillierter Ebene nicht komplett vorausplanbar. Stellenweise entscheidet sich erst durch Daten, die während des Entwurfs entstehen, wie weiter vorgegangen werden kann. Oft ist eine komplette Vorausplanung auch gar nicht erwünscht, da bestimmte Vorgehensweisen nicht von vornherein festgelegt werden sollen, sondern dem Bearbeiter vor Ort die Möglichkeit gegeben werden soll, das ihm am günstigsten erscheinende Vorgehen auszuwählen. Der Scheduler unterstützt diese Weiter- und Feinplanung, indem er eine Methodendefinition und -auswahl auch während der Bearbeitung zuläßt.

5.1. Unterstützung von Änderungsprozessen

Während des Entwurfsprozesses findet immer wieder Umplanung statt, Entscheidungen müssen rückgängig gemacht werden, weil sich z. B. Vorgaben ändern oder Konflikte entstehen. Durch die Rücknahme von Entscheidungen werden Assignments ungültig, die wiederum die Grundlage anderer Entscheidungen bilden. Durch das TMS können jetzt alle die Entscheidungen und Assignments bestimmt werden, die auf der gerade zurückgenommenen Entscheidung beruhen. Diese Entscheidungen müssen zwar nicht zwangsläufig ungültig werden, sie sind aber von der Änderung betroffen und sollten überdacht werden. Da im TMS die Assignments nur als ungültig markiert und nicht etwa gelöscht werden, ist es immer möglich, auf frühere Versionen zurückzugreifen. Wenn man eine frühere Entscheidung wieder gültig macht, bedeutet das allerdings nicht, daß damit auch automatisch alle Entscheidungen wieder gültig werden, die als Folge der Rücknahme ebenfalls ungültig wurden. Es ist zwar möglich, auch auf diese wieder zuzugreifen, bei einem Automatismus bestünde jedoch die Gefahr, daß sie mit in der Zwischenzeit erarbeiteten Ergebnissen in Konflikt kommen und diese daher ungültig machen würden.

6. Eine Anwendung: Bebauungsplanung

Im Projekt „Intelligenter Bebauungsplan“ (IBP) werden die bisher vorgestellten Techniken benutzt, um den Stadtplaner bei der Erstellung eines Bebauungsplans zu unterstützen und auch bei der weiteren Nutzung des Plans zu assistieren, z. B. wenn es darum geht, dem interessierten Bürger Auskunft zu geben.

Die Domäne

Gemeinden haben im Rahmen der Bauleitplanung die Aufgabe, die bauliche und sonstige Nutzung von Grundstücken vorzubereiten und zu leiten. Ein Teil davon besteht in der Erstellung

eines Bebauungsplans, der durch zeichnerische und textliche Festsetzungen Art und Maß der baulichen Nutzung von Gebieten innerhalb einer Gemeinde festlegt. So kann z. B. festgelegt werden, ob ein Baugebiet ein reines Wohngebiet ist (Art der baulichen Nutzung), wie groß die maximale Anzahl der Vollgeschosse ist und wie hoch die baulichen Anlagen sein dürfen (Maß der baulichen Nutzung). Die Bauleitplanung ist Teil der kommunalen Gesetzgebung und unterliegt daher auch besonderen Anforderungen im Hinblick auf gesetzliche Grundlagen, so muß etwa das Erstellungsverfahren ordnungsgemäß und nachvollziehbar ablaufen.

Der Einsatz von CoMo-Kit

Das Modell für den Entwurfsprozeß von Bebauungsplänen wurde mit Fachexperten aus den Gebieten „Baurecht“ und „Raum- und Umweltplanung“ erstellt. Da viele Daten grafisch eingegeben werden, wurde das Geographische Informationssystem (GIS) „Smallworld GIS“ angebunden, mit dem Geometrien für zeichnerische Festsetzungen am Bildschirm konstruiert werden. Desweiteren wird auch die Datenbank des GIS benutzt, um diesen Teil der Daten effizient zu verwalten.

Ein Anwendungsbeispiel

Das Szenario, das im Beispiel betrachtet werden soll, sieht wie folgt aus: Ein Gebiet wird

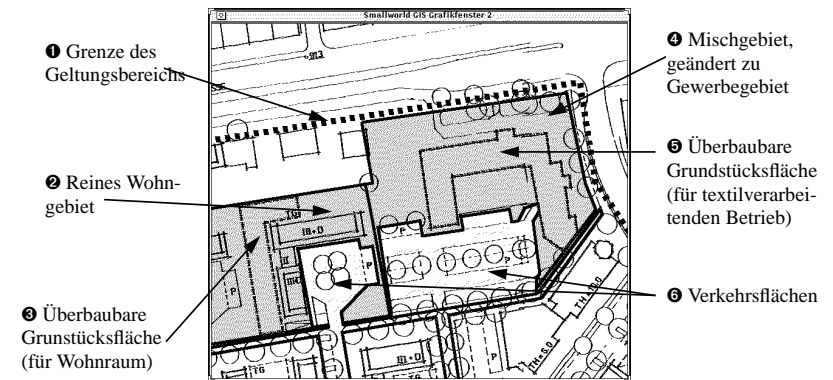


Abbildung 3: Das Beispielszenario, dargestellt im GIS

erschlossen, wie es im Informationsflußgraphen in Abbildung 4 zu sehen ist, die Planzeichnung und die dazugehörigen Erläuterungen sind in Abbildung 3 dargestellt. Nachdem die 1 Grenze des Geltungsbereichs festgesetzt worden ist, die das gesamte durch den Bebauungsplan behandelte Gebiet umschließt, wird das Gebiet 4 als Mischgebiet (Wohnen und bestimmte Arten von Gewerbe) ausgewiesen. Ein Investor möchte dort einen textilverarbeitenden Betrieb errichten. Dies darf er in einem bestimmten Umfang im Gebiet, das als 5 überbaubare Grundstücksfläche festgesetzt ist, tun. Dieses Gebiet wird für den Verkehr erschlossen 6. Aufgrund der zu erwartenden Lärmentwicklung wird ein Gutachten in Auftrag gegeben, das die Lärmbelastung untersucht und in dem für die geplante Nutzung eine noch tolerierbare Situation für

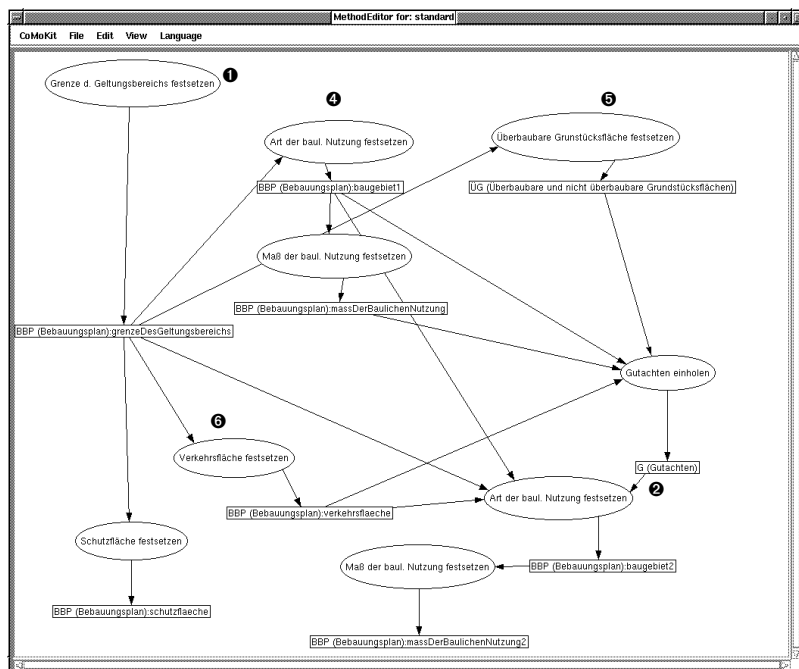


Abbildung 4: Der Informationsflußgraph für das Beispielszenario

angrenzende Gebiete feststellt. Deshalb kann auch das benachbarte Gebiet 2 als reines Wohngebiet ausgelegt werden.

Es besteht also ein Zusammenhang zwischen den Arten der Nutzung in den beiden Gebieten, der bei Änderungen beachtet werden muß.

Die Modellierung der Daten leitet sich aus dem Baugesetzbuch her und orientiert sich an dessen Paragraphen und Abschnitten.

Der Informationsfluß für das eingeführte Szenario wurde auf einer nicht zu detaillierten Ebene in Abbildung 4 dargestellt. Die einzelnen Aufgaben zerfallen noch in weitere Teilaufgaben, die wiederum in der Regel nur Teile der Ein- und Ausgaben betreffen. Für viele Festsetzungen gibt es generell die Möglichkeit, sie mit verschiedenen Vorgehensweisen zu treffen: sie können grafisch, textuell oder kombiniert erfolgen. Welche Möglichkeit gewählt wird, hängt dann in der konkreten Situation ab, wie sich eine Festsetzung am eindeutigsten beschreiben läßt.

Eine Änderung im Beispielszenario

Eine Änderung tritt ein, als der Investor sich stärker engagieren und den Betrieb ausweiten will. Insbesondere soll von diesem Standort her noch ausgeliefert werden. Dadurch ist stärkerer Fahrbetrieb zu erwarten, der Planer entschließt sich, die Nutzung des Gebiets zu einem Gewerbegebiet zu ändern. Die Änderung der Nutzung, mit der zunächst eine intensivere Nutzung des Gebiets einhergeht, hat eine stärkere Lärmbelastung für das angrenzende Wohngebiet zur Folge. Im Informationsfluß ist abzulesen, daß das Gutachten und die Art der baulichen Nutzung des Nachbargebietes zunächst direkt davon betroffen sind. Das Gutachten muß evtl. erneut eingeholt werden, aus ihm müssen Konsequenzen gezogen werden. Durch den erhöhten Fahrbetrieb werden die Grenzwerte für Lärm überschritten, der Planer hat das zu berücksichtigen, wenn er die Aufgabe „Art der baul. Nutzung festsetzen“ erneut bearbeitet. Da er dennoch die Nutzung von 2 als reines Wohngebiet aufrecht erhalten will, muß er Maßnahmen treffen, die die Lärmbelastung verringern, etwa indem er schalldämmende Fenster für die Gebäude vorschreibt (als zusätzliche textliche Festsetzung im Wohngebiet). Alternativ könnte er einen Lärmschutzwall angelegen (der sich aus der unterschiedlichen Nutzungen der beiden Baugebiete herleiten würde) oder die Lage der überbaubaren Grundstücksfläche verändern und die Hauseingänge an die lärmbelastete Seite und der Wohnbereich damit an die weniger laute Seite verlegen. Letzteres wäre ein Beispiel dafür, daß nicht automatisch alte Folgeentscheidungen gültig werden dürfen, wenn eine ursprüngliche Entscheidung wieder gültig wird. Selbst wenn der Investor sich wieder zurückzieht und damit das Gebiet 4 wieder Mischgebiet werden würde, muß deshalb nicht die Bauweise der Häuser im Wohngebiet wieder auf einen alten, evtl. nicht ganz ausgearbeiteten Stand zurückgesetzt werden.

7. Zusammenfassung und Ausblick

Mit dem hier vorgestellten Ansatz ist es möglich, Entwurfsprozesse auf einem hohen Detaillierungsgrad umfassend zu modellieren. Der dadurch entstandene Plan ist die Grundlage für die verteilte Abwicklung durch ein Workflow-Management-Werkzeug. Dabei wird das Um- und Weiterplanen während der Abarbeitung unterstützt, die Auswirkungen von Änderungen werden effizient verwaltet.

Weiterentwicklungen betreffen die Möglichkeit, oft und gleichartig auftretende Vorgehensweisen in Form von Prozeduren abzulegen, wodurch eine verbesserte Wartbarkeit des Modells erzielt wird.

8. Literatur

- (AAAI92) AAAI '92 Workshop on *Design Rationale Capture and Use*, July 15, 1992, San Jose, CA.
- (BSH94) S. Buckingham Shum and N. Hammond. Argumentation-based design rationale: what use at what cost? *Int. J. Human-Computer Studies*, 40:603–652, 1994.

- (CoB88) Conklin, J. and Begeman, gIBIS: A Tool for Exploratory policy Discussion, In *Proceeding of CSCW '88 and ACM Transactions on Office Information Systems*, October, 1988.
- (DMP95) B. Dellen, F. Maurer, J. Paulokat: Verwaltung von Abhängigkeiten in kooperativen, wissensbasierten Arbeitsabläufen. In *Proceedings of the 3^{ten} deutschen Expertensystemtagung*, pages 72-89, 1995
- (Doy79) Doyle, J.: A Truth Maintenance System, *Artificial Intelligence*, 12:231-272, 1979.
- (GeH95) Georgakopoulos, D. and Hornick, M.: An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure, *Distributed and Parallel Databases*, 3, pages 119-153, Kluwer Academic Publishers, Boston 1995
- (Jab95) Jablonski, S. Workflow-Management-Systeme: Motivation, Modellierung, Architektur: From Informatik Spektrum 18: pages 13-24, Springer-Verlag 1995
- (Obe94) Oberweis, A.: Verteilte betriebliche Abläufe und komplexe Objektstrukturen: Integriertes Modellierungskonzept für Workflow-Managementsysteme, Habilitation Universität Karlsruhe, 1994
- (MaL91) A. Mac-Lean et al. Questions, Options and Criteria; Elements of Design Space Analysis. *Human-Computer Interaction*, 6: 201-250, 1991.
- (Mau93) Maurer, F.: Hypermedia-Based Knowledge Engineering for Distributed Knowledge-Based Systems (in german), Dissertation Universität Kaiserslautern, 1993, auch: DISKI 48, infix-Verlag, ISBN 3-929037-48-3.
- (MPa94) Maurer, F., Paulokat, J.: Operationalizing Conceptual Models Based on a Model of Dependencies, in: A. Cohn (Ed.): *ECAI 94*. 11th European Conference on Artificial Intelligence, 1994, John Wiley & Sons, Ltd.
- (MPe95) Maurer, F., Pews, G.: Ein Knowledge-Engineering-Ansatz für kooperatives Design am Beispiel der Bebauungsplanung, Themenheft Knowledge Engineering, KI 1/95, interdata Verlag, 1995.
- (Pet91) Petrie, Ch.: Planning and Replanning with Reason Maintenance, Dissertation, University of Texas, Austin, 1991.
- (PeC94) C. Petrie and M. Cutkosky. Design space navigation as a collaborative aid. In J. Gero and F. Sundweeks, editor, *Artificial Intelligence in Design '94*, Kluwer Academic Publishers, 1994
- (PoB88) C. Potts and G. Bruns, Recording the Reasons for Design Decisions, In *Proceedings of the 10th International Conference on Software Engineering*, pages 418-427, 1988.
- (SDR91) Special Issue on Design Rationale, *Human-Computer Interaction*, 6: 197-419, 1991.