

InteGrA - EINE INTEGRIERENDE GROUPWARE-ANWENDUNG FÜR EIN ARCHITEKTURBÜRO

Christian Müller

Robert G. Rodewald

Institut für Industrielle Bauproduktion - ifib, Universität Karlsruhe (TH)

mailto:christian.mueller@ifib.uni-karlsruhe.de

http://www.ifib.uni-karlsruhe.de

Kühnl & Schmidt, Architekten, Karlsruhe

kuehnl@t-online.de

Die speziellen Charakteristika der Bauplanung sowie die derzeit schwierigen wirtschaftlichen Rahmenbedingungen, stellen Architektur- und Ingenieurbüros vor große Herausforderungen bezüglich einer effektiveren Unterstützung ihrer Planungsprozesse. Dieser Artikel beschreibt die Ergebnisse eines Kooperationsprojektes, bei dem das Institut für Industrielle Bauproduktion in Zusammenarbeit mit einem Architekturbüro eine integrierende computergestützte Planungsumgebung entwickelt und in diesem Unternehmen eingeführt hat. Ziele waren eine QM-gerechte Projektdokumentation, aktive Projektmanagementunterstützung, der Aufbau von Wissenspools sowie die Ermöglichung eines standortunabhängigen Arbeitens. Am Anfang standen umfangreiche Arbeiten zur Untersuchung und Systematisierung interner und externer Kooperationsprozesse. Die dort entwickelten Konzepte wurde dann softwaretechnisch über eine Reihe spezifischer Datenbankanwendungen auf Basis der Groupwareplattform Lotus NOTES® umgesetzt. Auf verschiedenen Ebenen wurden Integrationsmechanismen entwickelt, um von der überwiegend isolierten hin zu einer standortunabhängigen kooperativen Rechnernutzung zu kommen.

1 Einleitung

Dieser Beitrag beschreibt die Ergebnisse eines einjährigen Kooperationsprojektes des Instituts für Industrielle Bauproduktion (ifib) mit dem Architekturbüro Kühnl & Schmidt, in dessen Rahmen eine integrierende Planungsumgebung für das Architekturbüro auf Basis der Groupwareplattform Lotus NOTES® entwickelt wurde. Kühnl & Schmidt ist ein auf zwei Standorte (Karlsruhe und Leipzig) verteiltes Architekturbüro, das konsequent auf den Einsatz von Informationstechnologie zur Unterstützung ihrer Planungsprozesse setzt. Angestrebtes Ziel ist der Übergang der noch überwiegend einzelbenutzerbezogenen Rechnernutzung für CAD, Office, AVA, Graphik, etc. hin zu einer standortunabhängigen kooperativen Nutzung zur Unterstützung von Kommunikation, Koordination, Kooperation und Informationsmanagement.

Die Motivation zu diesem Projekt ergab sich vor allem aus der Tatsache, daß verfügbare Produkte nur unzureichend den Anforderungen an Organisation, IT-Infrastruktur und Planungspro-

zesse in diesem Büro entsprachen. Die Idee war deshalb, eine spezifische, aber dennoch offene Planungsumgebung zu schaffen, die vom Architekturbüro intern selbst erweitert und den Veränderungen entsprechend angepaßt werden kann. Das Kooperationsprojekt wurde deshalb so ausgelegt, daß der Industriepartner nicht nur in die Konzeptphase, sondern sukzessiv auch in die Implementierungsarbeiten am ifib miteinbezogen wurde, um nach Ende des Projektes das notwendige Know-How zur Weiterentwicklung des Systems zu besitzen. Zu Projektbeginn wurden folgende grobe Zielvorgaben definiert:

- Verbesserung der internen und standortübergreifenden Kommunikation sowie eines reibungslosen Informationsflusses, möglichst auch unter Einbeziehung externen Fachplaner
- Unterstützung des Projektmanagements durch eine strukturierte, QM-gerechte und rechnergestützte Projektdokumentation und aktive Hilfestellung über elektronische Checklisten
- Aufbau spezieller aspektbezogener Wissenspools. Dabei sollte ein im Unternehmen schon vorhandenes Konzept weiterentwickelt und in eine rechnergestützte Anwendung überführt werden.

Die teamorientierte Entwicklung basierte auf einem iterativen Vorgehen über den Entwurf und die Implementierung von Prototypen, die man sukzessive bewerten und anpassen konnte.

Nach eingehender Analyse verfügbarer Groupwareumgebungen entschied man sich für die Anwendung des Lotus NOTES® domino Servers und der NOTES Clients. NOTES ist kein Produkt mit festem anwendungsspezifischen Funktionsumfang, sondern eine leistungsfähige Entwicklungsumgebung, die eine ganze Reihe an Basisfunktionalitäten von Groupware schon bereitstellt. Auf diese Plattform können maßgeschneiderte Anwendungen aufgesetzt werden. Die Struktur von NOTES basiert auf dokumentorientierten Datenbanken, in der die gesamte Information in einzelnen NOTES Dokumenten gespeichert wird. Diese Dokumente werden über Masken - eine Art Formular - erzeugt und dargestellt. Dabei kann jede Maske Felder unterschiedlichen Typs, OLE-Objekte, JAVA Applets oder auch Programmcode enthalten. Ebenso können Datenbanken und einzelnen Feldern Programmcode zugeordnet werden. Eine Darstellung der gesamten in der Datenbank gespeicherten Information (Dokumente) erfolgt über frei definierbare Ansichten ("views"). Für eine detailliertere Erläuterung von NOTES wird auf die umfangreich existierende Fachliteratur verwiesen.

2 Systemstruktur

Ganz am Anfang standen Überlegungen, wie man die geforderten Funktionalitäten auf verschiedene NOTES Datenbanken (NDB) aufteilen kann. Dazu sind eine Reihe technologischer Aspekte wie z.B. Größenbeschränkung von NDBen, Replikationsmechanismen, Reichweite von Programmierfunktionen als auch eine von der Struktur der Anwendung abgeleitete natürliche Kapselung zu berücksichtigen. Man entschied sich für folgende Aufteilung:

Für jedes neue Projekt wird aus einer speziellen Schablone eine strukturgleiche Datenbank erzeugt, die die gesamte für das Projekt relevante Information aufnimmt und verwaltet. Die korrekte Erzeugung und Verwaltung dieser Projektdatenbanken wird durch eine spezielle Anwen-

dung, der Projektübersichtsdatenbank, ermöglicht und unterstützt. Vorteile dieser projektbezogenen Kapselung sind vor allem eine projektbezogene Benutzerverwaltung, die selektive Replikation und Archivierung einzelner Projekte und die Tatsache, daß die Anzahl der verwalteten Projekte nur vom verfügbaren Festplattenspeicher begrenzt wird.

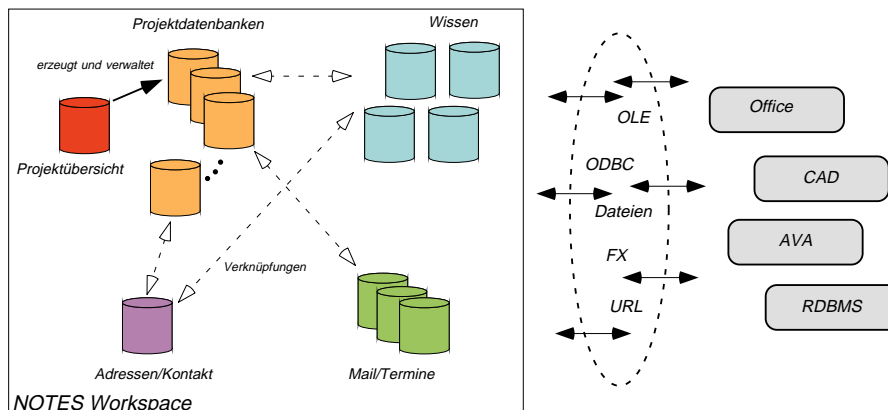


Abb. 2.1: Aufteilung in Datenbanken und Integration externer Anwendungen

Planungswissen wird zwar zu großen Teilen innerhalb der Projektarbeit erworben, sollte aber projektübergreifend allen Mitarbeitern im Unternehmen explizit zur Verfügung stehen. Die Art der Aufbereitung des Wissens hängt jedoch sehr stark von der Anwendungsdomäne ab. Somit ergibt sich die

Aufteilung des "Wissenspools" in eine Kollektion von Datenbanken mit unterschiedlicher Struktur. Im Rahmen des Projektes wurden im Interessensgebiet "Ökologisches Bauen" vier aspektbezogene und strukturähnliche Datenbanken entworfen und implementiert. Adressen werden standortübergreifend in einer separaten Datenbankanwendung verwaltet. Zur Kommunikation und zur Terminvereinbarung werden die von NOTES angebotenen Maildatenbanken genutzt. Trotz der Kapselung in unterschiedliche Datenbanken können diese bei Bedarf geeignet verknüpft werden. NOTES bietet verschiedene Möglichkeiten einer datenbankübergreifenden Integration an. Einfache Beispiele hierfür sind z.B. die Übernahme von Adreßfeldern aus der AdreßDB in eine ProjektDB, ein Link aus der Wissensdatenbank auf ein Planungsbeispiel oder die Nutzung der E-mail Datenbanken zur Notifikation eines Workflow-Vorgangs. Abbildung 2.1 zeigt schematisch die Aufteilung in die verschiedenen Datenbanken und die Integrationsmechanismen externer Anwendungen in die Planungsumgebung. Man entschied sich hier für eine fallweise Einbindung über OLE, FieldExchange (FX) - eine NOTES spezifische Übernahmemöglichkeit von Feldinhalten - sowie über URL und Fileattachments.

Im folgenden beschränkt sich der vorliegende Artikel auf die Erläuterung der Projektbanken, die den wesentlichen Entwicklungsaufwand darstellten und gleichsam das Herzstück des Systems bilden.

3 Projektdokumentationsanwendung

Ziel des Projektdokumentationswerkzeugs ist eine assistierte und QM-gerechte Erstellung und Verwaltung der gesamten projektbezogenen Information. Verschiedene Sichten auf das Projekt, einfache Bedienung und Transparenz bezüglich der erzeugten Information aber auch des Projektstandes waren weitere Anforderungen. Darüberhinaus sollte es auch passive und aktive Unterstützung beim Projektmanagement bieten. Das Projektdokumentationswerkzeug wird

somit zur wichtigsten projektbezogenen Arbeitsumgebung des Benutzers, die alle anderen Anwendungen möglichst nahtlos integriert. Im Idealfall ist keine Benutzung der Betriebssystemoberfläche über den Login-Vorgang hinaus mehr notwendig. Alle benötigten Applikationen werden aus der Arbeitsumgebung heraus gestartet und legen ihre Daten direkt an der vorgesehenen Stelle in der Projektdatenbank ab. Die Integrationsmechanismen wurden schon im vorherigen Kapitel (Abb. 2.1) angedeutet. Das System sollte sich möglichst gut an den Abläufen und der Organisation im eigenen Unternehmen orientieren, aber auch eine Einbeziehung externer Fachplaner ermöglichen.

3.1 Vorgehen bei der Entwicklung

Am Anfang stand eine eingehende Analyse des gesamten Planungsprozesses im Architekturbüro, vom ersten Bauherrengespräch bis hin zur Archivierungsverpflichtung der gesamten Projektdokumentation. In regelmäßigen Abständen wurden auch alle Mitarbeiter des Architekturbüros in diesen Analyseprozeß aktiv miteinbezogen. Aufbauend auf diese Ergebnisse erfolgte dann eine Systematisierung der Informationen und der Abläufe, auch schon im Hinblick auf Implementierungsmöglichkeiten in NOTES. Dabei wurde der HOAI-Planungsprozeß in vier große Phasenblöcke aufgeteilt und sequentiell systematisiert. Man konnte somit bereits nach der Aufbereitung des ersten Phasenblocks parallel zu weiteren Konzeptionsarbeiten mit der Implementierung eines ersten Prototypen beginnen. Anhand der dadurch erhaltenen Erkenntnisse konnten viele Umsetzungsprobleme rechtzeitig erkannt und in die weiteren Definitionsarbeiten miteinbezogen werden.

Der Implementierung liegt in weiten Teilen die Vorstellung des *Containermodells* zugrunde. Informationen werden dabei nicht direkt im Dateisystem gespeichert, sondern in einem komplexen Containerelement abgelegt, das die Möglichkeit bietet, zusätzliche Information (sog. Metainformation) zum eigentlichen Containerinhalt zu verwalten. Diese Metainformation kann dann vom Datenbanksystem genutzt werden, um spezifische Sichten auf die Information zu generieren, Zugriffe zu regeln, Kontexterläuterungen zu verwalten oder aktive Prozeßunterstützung zu bieten. Der Container kann idealerweise beliebige Information aufnehmen, unabhängig von Art und Format. Somit erreicht man eine flexible, aber dennoch systematische Erfassung der Information. Im folgenden wird das generelle Vorgehen der Entwicklung und deren wichtigste Elemente sequentiell beschrieben. Diese strenge Reihenfolge entsprach bei der sukzessiven Entwicklung über Prototypen natürlich nicht ganz der Realität, schafft hier aber größere Transparenz.

3.2 Implementierung

Wie im ersten Kapitel bereits erwähnt, wird die gesamte Information in NOTES in Form von sogenannten *Dokumenten* gespeichert und dargestellt. Notesdokumente stellen somit im oben skizzierten Containermodell ein Containerelement dar. Deshalb erfolgte im ersten Schritt die statische Abbildung der benötigten Information auf die NOTES Elemente *Masken*, *Teilmasken*, *Profildokumente*, die dazugehörigen *Felder* sowie auf *Ansichten*. Dabei ist durch geschicktes Design die Anzahl verschiedenartiger Masken möglichst gering zu halten und die Wiederverwendbarkeit über Teilmasken zu maximieren.

Im Rahmen dieses Artikels können die einzelnen Masken natürlich nicht detailliert vorgestellt werden. Deshalb wird nur kurz der generelle Aufbau skizziert. Alle verwendeten Masken enthalten mehrere oder alle der nachfolgend genannten Teile:

- **Systemparameter:** Dieser Teil des Dokumentes ist für den Benutzer immer unsichtbar und enthält vom System ausgewertete Parameter.
- **Dokumentinformation:** Die hier definierten Felder enthalten kategorisierende, beschreibende und inhaltlich relevante Prozeßinformation. Einige Felder werden automatisch ausgefüllt, andere hingegen können vom Benutzer bearbeitet werden. Beispiele hierfür sind Dokumententitel, Freigabe, Herkunft und Gewerkezuordnung. Zur Repräsentation der Dokumente in den Ansichten wird überwiegend die Information dieses Teils verwendet.
- **Inhaltliche Information:** Hier wird die eigentliche inhaltliche Planungsinformation gehalten. Die Felddefinitionen dieses Teils variieren zum Teil sehr stark von Maske zu Maske. Dennoch findet sich neben weiteren Feldern hier fast immer ein Feld des Typs rtf (rich text format), das als *Informationscontainer* bezeichnet wird. Dieses rtf-Feld kann binäre Objekte beliebigen Typs (OLE, Dateien beliebigen Formats, Graphiken, ...) aufnehmen.
- **Logbuch:** Der Benutzer kann hier das Logbuch aller protokollierten Aktionen bezüglich dieses Dokumentes einsehen (Abbildung 3.2 unten). Einträge werden vom System bei verschiedenen Aktionen über eine spezielle Prozedur erstellt.
- **WorkFlow:** Diese Teilmaske bildet die Benutzerschnittstelle zur implementierten WorkFlow-Funktionalität. Diese wird im folgenden noch näher erläutert.

Abbildung 3.1 zeigt an einem Beispiel den Dokumentinformationsteil, sowie den Inhaltsteil mit assistierter Eingabe des Plancodes und einem AutoCAD OLE-Objekt im Informationscontainer.

Zugriffsregeln auf Dokumente und einzelne Funktionalitäten, werden über Rollen und die Modifikation spezieller Autorenfelder in Dokumenten erreicht, die auf die Systematik der in NOTES verwendeten Zugriffskontrollliste (ACL) abgebildet werden. Dabei geht man von einem einfachen, aber sehr flexiblen Modell aus: Es gibt drei verschiedene Rollen, die entsprechenden Benutzern beliebig zugeordnet werden können: *Planer*, *Projektleiter* und *DB-Admin*. Der *DB-Admin* legt eine Projektdatenbank an, vergibt die Rollen und kann beliebige Änderungen an der Struktur der DB vornehmen. Der *Projektleiter* hat Bearbeitungsrechte auf alle Dokumente und kann spezielle Aktionen der aktiven Projektmanagementunterstützung (Projektprofile, Checklisten,...) nutzen. Der *Planer* hat lesenden Zugriff auf alle und Bearbeitungsrechte nur auf die von ihm erzeugten Dokumente. Die Bearbeitungsrechte auf ein Dokument können

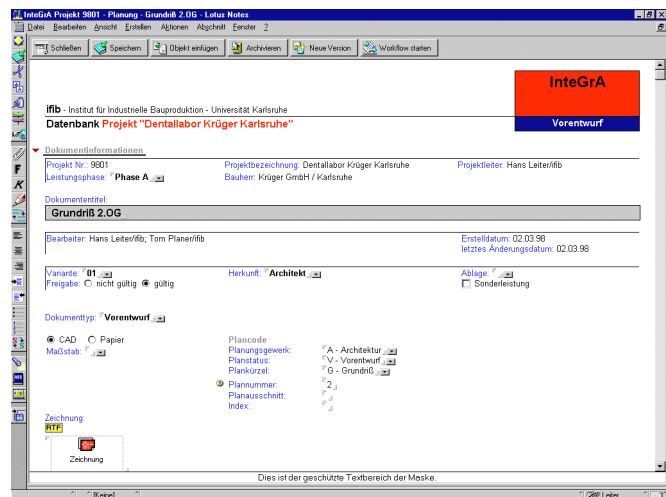


Abb. 3.1: Teilansicht einer Maske

allerdings fallweise von dessen Erzeuger auf eine beliebige Anzahl anderer Benutzer ausgedehnt und auch wieder eingeschränkt werden. Hierzu wird die im folgenden noch erläuterte Workflow-Funktionalität verwendet.

Nach einer ersten Definition der Masken erfolgte dann im zweiten Schritt die Implementierung der eigentlichen Systemfunktionalitäten. Aus der Vielzahl sollen hier nur einige wenige skizziert werden:

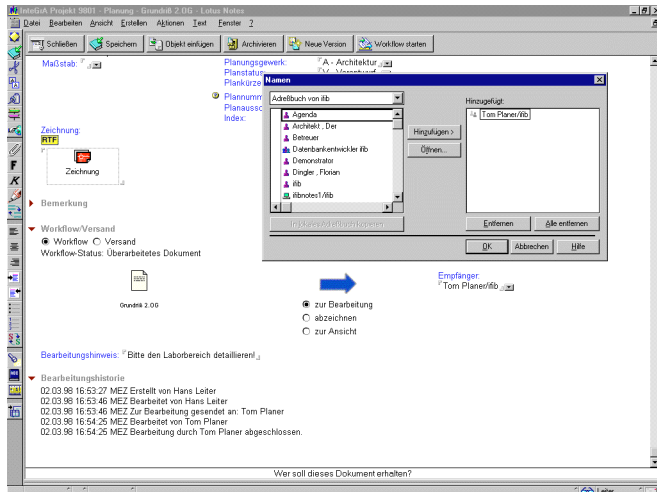


Abb. 3.2: Teilmaske Workflow

WorkFlow: Um die Kooperation in den schwach strukturierten Planungsprozessen zu unterstützen, wurde eine einfache und flexible Ad-Hoc WorkFlow Komponente entwickelt. Zur Information über WorkFlow Aktivitäten nutzt sie die Maildatenbanken der Benutzer. Drei verschiedene Vorgänge sind implementiert:

1. *Zur Ansicht:* Dieser Vorgang schickt an eine beliebige Anzahl Benutzer eine Nachricht mit eingefügtem Link auf das spezifische Dokument. Dies wird beispielsweise dazu benutzt, um Teamkollegen auf sie

betreffende Änderungen im Informationsbestand hinzuweisen. Bearbeitungsrechte werden dabei nicht geändert.

2. *Zur Bearbeitung:* An einen vom Sender frei definierbaren Benutzer wird ein Bearbeitungshinweis mit angefügtem Link auf das Dokument geschickt. Diesem Empfänger werden Bearbeitungsrechte auf das Dokument zugewiesen. Das Dokument bleibt solange als "im Workflow" gekennzeichnet, bis der Empfänger den Abschluß seiner Bearbeitung bestätigt und dadurch auch seine Bearbeitungsrechte verliert. Der Sender wird hiervon automatisch unterrichtet. So können kooperative Planungsvorgänge mit dem Sender als koordinierende Instanz unterstützt werden. Der Sender hat jederzeit die Möglichkeit, den Workflow abubrechen und erhält die alleinigen Bearbeitungsrechte zurück.

3. *Dokument abzeichnen:* Einer beliebigen Anzahl weiterer Benutzer werden Dokumentverweise mit der Aufforderung zum Abzeichnen geschickt. Diese erhalten Bearbeitungsrechte auf das Dokument, um z.B. Bemerkungen hinzuzufügen. Haben alle Empfänger das Dokument mit ihrer elektronischen ID abgezeichnet, erhält der Sender eine automatische Bestätigung. Diese Bestätigung entzieht dem Benutzer wiederum seine Bearbeitungsrechte. Mit dieser Funktionalität werden Abzeichnungsvorgänge, aber auch vom Sender nicht explizit koordinierte Teamprozesse unterstützt.

Der Benutzer kontrolliert den WorkFlow über ein leicht verständliches Interface (Abbildung 3.2). Workflow-Aktionen wie z.B. *WorkFlow-Starten*, *WorkFlow-Abbrechen*, *Abzeichnen* oder *Bearbeitung bestätigen* werden über Aktionsbuttons gesteuert, die empfangen- und rollenabhängig eingeblendet werden.

Checklisten: Um einen schnellen Überblick über den Projektstand durch Anzeige der schon abgeschlossenen und noch zu erledigenden Aufgaben zu ermöglichen, wurden spezielle phasenbezogene Checklisten entwickelt. Deren Besonderheit ist, daß Einträge überwiegend nicht beliebig manuell, sondern nur vom Projektleiter über gültige und dem Eintrag korrespondierende Informationen bestätigt werden können. Darüberhinaus können weitere, frei definierbare Bedingungen überprüft werden. Checklisten sind mit den Projektprofilen in der Ansicht "Überblick" zu finden.

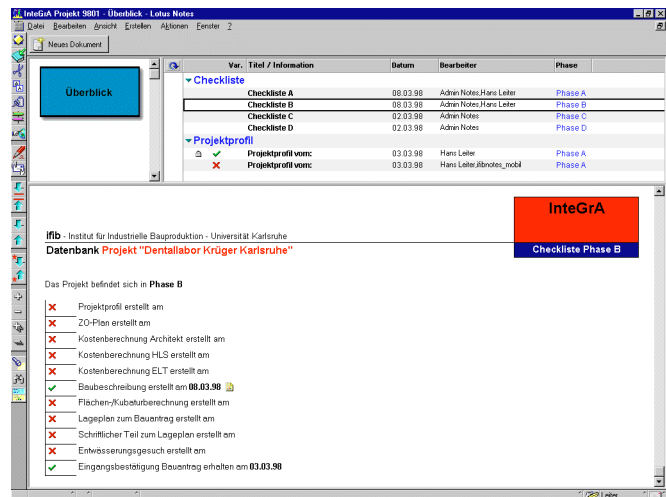


Abb. 3.3: Checklistendokument der Phase B

Projektprofile: Projektprofile werden genutzt, um Anforderungen frühzeitig und systematisch über die Projektphasen zu erfassen und festzuschreiben. Diese wichtigen Grundlagen der Projektarbeit können dem Planungsteam somit transparent zur Verfügung gestellt werden. Weiterhin lassen sich dadurch Entwicklungen im Projekt sehr gut dokumentieren.

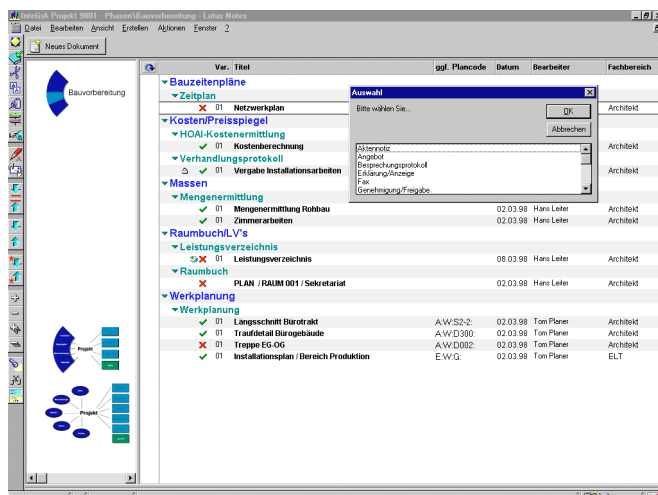


Abb. 3.4: Aktion "neues Dokument"

ermöglicht über ein Auswahlfenster nur die Erzeugung der in dieser Phase gültigen Informationen.

Den Aufruf der vielfältigen Ansichten erfolgt über das Design graphischer Navigatoren. Über diese werden die Ansichten aufgerufen und stellen eine intuitive graphische Benutzerschnittstelle dar. Die Analyse der Arbeitsweisen im Architekturbüro K&S hat gezeigt, daß zwei Hauptnavigatoren nötig sind. Der erste ist nach HOAI-Phasenpaketen (Abbildung 3.5) und der andere nach Typ der Information gegliedert. Zwischen beiden Navigatoren kann beliebig hin-

Im nächsten Schritt erfolgte die Definition der Ansichten, die als eine Art Filter spezielle tabellarische Sichten auf den gesamten Datenbestand bereitstellen. Dabei werden dem Benutzer innerhalb dieser Ansichten auch nur die jeweils sinnvollen und dort gültigen Aktionen angeboten. Man kann somit nicht nur Information kontextuell bezogen zusammenfassen, sondern erreicht auch eine vereinfachte Bedienung durch die notwendige Kapselung der Gesamtfunktionalität. Abbildung 3.4 zeigt eine solche Ansicht. Die Aktion *Neues Dokument*

und hergeschaltet werden. Jeder der beiden Navigatoren bietet Zugriff auf die gesamte Systemfunktionalität.

Wie schon in Kapitel 2 erwähnt, werden die einzelnen Projektdatenbanken durch eine spezielle Anwendung - die Projektübersichtsdatenbank - erzeugt und verwaltet. Für die Planer bietet sie eine einheitliche Ansicht über alle Projekte. Sie stellt ihm allgemeine Informationen, die Rollenvergabe und den aktuellen Projektstatus dar, dient aber auch gleichzeitig als Navigationshilfe für das entsprechende Projekt. Für den ProjektDB Admin hingegen ist es das zentrale Werkzeug zur assistierenden Erzeugung der Datenbanken. Es übernimmt dabei auch konsistente Änderungsvorgänge der Rollenvergabe und Projektinformationen während des Projektverlaufs.

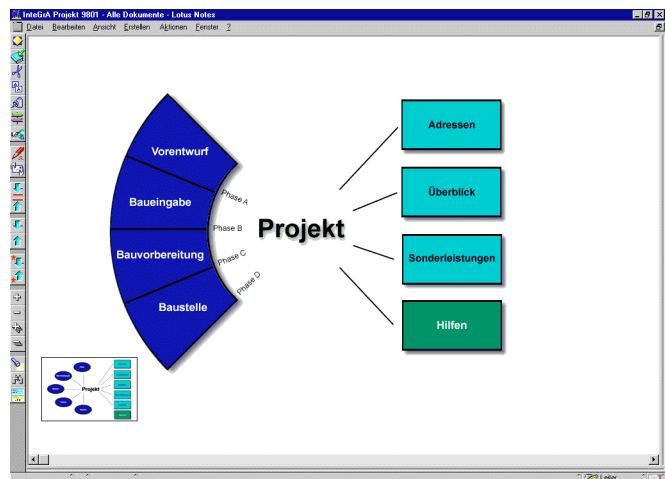


Abb. 3.5: Hauptnavigator nach Phasen

4 Zusammenfassung / Ausblick

Der vorliegende Artikel gab einen Überblick über das Ergebnis des einjährigen Kooperationsprojektes des ifib mit einem Architekturbüro. Dieses hatte die Entwicklung der ersten Version einer an den Anforderungen des Unternehmens orientierten und mitwachsenden Planungsumgebung zum Ziel. Beide Kooperationspartner sind davon überzeugt, daß nur diese offenen und sich an den spezifischen Gegebenheiten orientierenden Systeme den gewünschten Fortschritt bei wirklichen integralen Planungsprozessen darstellen. Ein Schlüsselkriterium für den erfolgreichen Einsatz des hier vorgestellten Planungssystems ist die Möglichkeit, es den veränderten Anforderungen flexibel anpassen zu können. Die hier vorgestellten spezifischen Groupwareanwendungen stellen deshalb nur eine Momentaufnahme in ihrer Entwicklung dar. Schon aufgrund der begrenzten Entwicklungszeit konnten einige der während der Kooperation entwickelten Ideen und Vorstellungen noch nicht in Systemfunktionalität umgesetzt werden. Durch die enge Kooperation und den damit verbundenen Know-How Transfer ist das Architekturbüro nun in der Lage, das System konzeptuell und funktional eigenständig weiterzuentwickeln. Neben reinen funktionalen Erweiterungen (wie z.B. aktive Controlling-Werkzeuge, intelligente Raumbuchfunktionen, etc.) sehe ich im wesentlichen zwei weiterführende Schwerpunkte, die starke Abhängigkeiten haben:

Zum einen die Erweiterung des strukturellen Konzeptes zur besseren Einbindung externer Planungspartner und zum anderen die Anstrengungen bezüglich einer projektspezifischen Anpassbarkeit der Groupwareanwendungen. Diese beiden Punkte adressieren aktuelle Forschungsbemühungen in den Bereichen Organisations- und Managementmodelle virtueller Organisationen und die partizipative Anpassbarkeit von Groupware.