

Ansatz einer prozeßzentrierten Produktentwicklungsumgebung am Beispiel der Angebotsbearbeitung in Gießereien

Martin Endig^{*}, Sylke Kröttsch[†], Dirk Jesko^{*}, Georg Paul^{*}

^{*}Institut für Technische und Betriebliche Informationssysteme, Otto-von-Guericke-Universität Magdeburg, Postfach 4120, D-39016 Magdeburg
{endig, jesko, paul}@iti.cs.uni-magdeburg.de

[†]Institut für Fertigungstechnik und Qualitätssicherung, Otto-von-Guericke-Universität Magdeburg, Postfach 4120, D-39016 Magdeburg
kroetz@iti.cs.uni-magdeburg.de

Zusammenfassung

Die Verwendung von modernen informationstechnischen Konzepten und Methoden gewinnt für die Ausführung von Produktentwicklungsprozessen für kooperierende Unternehmen zunehmend an Bedeutung. Zur Ausführung dieses Prozesses kommen eine Reihe von unterschiedlichen Softwarewerkzeugen zum Einsatz, die durch die Bereitstellung innerhalb spezieller Entwurfsumgebungen unternehmensweit flexibler eingesetzt werden können. Daraus ergibt sich nicht nur die Forderung nach einem Integrationskonzept für heterogene Softwarewerkzeuge, sondern auch nach einem Konzept zur Unterstützung der auszuführenden Entwicklungsprozesse. Im Rahmen dieses Beitrages soll ein Ansatz für eine prozeßzentrierte Produktentwicklungsumgebung am Beispiel der Produktionsvorbereitung vorgestellt werden, die die auszuführenden Entwicklungsprozesse in den Mittelpunkt der Betrachtungen stellt. Darüber hinaus wird ein Prototyp vorgestellt, mit welchem die Umgebung validiert wurde.

1 Einleitung

Aufgrund des ständig wachsenden Konkurrenzkampfes auf dem Weltmarkt sind die Unternehmen des Maschinen- und Anlagenbaus in Deutschland gezwungen, ihre Prozesse der Produktentwicklung und -fertigung durch einen besseren Einsatz von technischen und organisatorischen Hilfsmitteln zu rationalisieren [22]. Dieses wird beispielsweise durch die Einführung von neuen Produktionsorganisationskonzepten oder durch die Reduzierung der Fertigungstiefen innerhalb der Unternehmen erreicht. Dadurch ist es möglich, die Produktentwicklungszeiten für komplexe Produkte wesentlich zu verkürzen und darüber hinaus die Qualität der Produkte bei gleichzeitig sinkenden bzw. gleichbleibenden Kosten deutlich zu erhöhen. Zur Realisierung dieser gegensätzlichen Ziele ist der Einsatz von modernen Methoden und Techniken der

Informatik in allen technologisch notwendigen Entwicklungsphasen unabdingbar. Der auszuführende Prozeß wird im allgemeinen als *Engineeringprozeß* bezeichnet. Dieser umfaßt alle technologischen Prozesse, die für die Entwicklung, Fertigung und Wartung eines Produktes notwendig sind. Der Engineeringprozeß ist besonders durch häufige Benutzerinteraktionen gekennzeichnet und beinhaltet gemäß [20] Prozesse, die von der Produktidee bis zur Markteinführung eines Produktes reichen. Im folgenden sollen die Produktentwicklungsprozesse (zum Beispiel Produktplanung, Konstruktion oder Arbeitsplanung) näher betrachtet werden. Diese Prozesse sind zum einen durch die Verkettung von kreativen, zumeist nicht sequentiellen Entwicklungs- und Gestaltungsschritten und zum anderen durch den Aufbau, die Analyse und die Modifizierung des *Produktstrukturmodells* in Teamarbeit gekennzeichnet [29]. Für letzteres ist eine rechnerinterne, integrierte Abbildung von Entwicklungsprozessen *und* Produktmodellen notwendig. Dabei werden verschiedene Methoden und Werkzeuge u.a. zum Entwurf und zur Simulation von Produktteilen mit dem Ziel der Herstellung von Fertigungsunterlagen eingesetzt. Diese Prozesse werden auch als *Entwurfsprozesse* bezeichnet.

Moderne Methoden und Techniken der Informatik zur Realisierung der Ausführung von Entwurfsprozessen unterstützen in der Regel nur einzelne Aspekte bzw. Teilprozesse direkt. Hierzu zählen beispielsweise neben spezialisierten Softwarewerkzeugen, wie CAD-Systeme zur Konstruktion von Bauteilen, auch *Engineering Data Management Systeme* (EDM-Systeme), mit welchen auf der Basis eines Produktstrukturmodells die einheitliche Verwaltung der Produktinformationen über den gesamten Lebenszyklus eines Produktes möglich ist [13]. Die für die Erzeugung der Produktinformationen erforderlichen spezifischen Werkzeuge werden über proprietäre Schnittstellen an die EDM-Systeme gekoppelt, wobei dann die permanente Speicherung der Informationen durch diese Systeme realisiert werden kann.

Die durchgängige Unterstützung von Entwurfsprozessen sowohl durch prozeßspezifische als auch durch prozeßübergreifende Softwarewerkzeuge ist noch nicht ausreichend [31]. So ist es beispielsweise mit den vorhandenen Systemen kaum möglich, geschäftsprozeßorientierte Strukturen innerhalb der Produktentwicklung einzuführen, um eine ganzheitliche und vor allem einheitliche Abbildung der auszuführenden Geschäfts- oder auch Entwurfsprozesse zu erreichen. Derzeitig werden u.a. die Produkt- und die Lieferanteninformationen verwaltet, nicht jedoch der eigentliche Prozeß zur Entwicklung eines Produktes. Dieses ist notwendig, um u.a. sowohl die Vergabe von Teilaufträgen an andere Unternehmen als auch die dezentrale Planung und Fertigung von Produktteilen zu organisieren. In diesem Zusammenhang gewinnt neben der *Produktdatenverwaltung* (realisiert durch EDM-Systeme) auch die *Prozeßverwaltung* zunehmend an Bedeutung [14]. Diese wird aber von heutigen EDM-Systemen nicht adäquat bereitgestellt [2]. Auch der Einsatz von Workflow-Technologien bietet aufgrund der fehlenden Ausrichtung entsprechender Systeme auf den Engineeringbereich keine befriedigende Lösung an.

Um die Entwurfsprozesse innerhalb der Unternehmen besser unterstützen zu können, ist die Einführung *erweiterter* Technologien notwendig, die aufbauend auf den vorhandenen Softwarewerkzeugen innerhalb eines Unternehmens zusätzliche Funktionalitäten anbieten. Spezielle *Entwurfsumgebungen* oder *CAX-Umgebungen* können dieses leisten. Entwurfsumgebungen können beispielsweise die Arbeit eines Kon-

strukteurs unterstützen, indem sie eine Menge von Werkzeugen bereitstellen, die eine einheitliche Bearbeitung aller während der Entwicklung anfallenden Entwurfsdokumente ermöglichen. Sie stellen aus Sicht des Anwenders „nur“ eine Arbeitsumgebung dar, die Daten und Entwurfsdokumente verwalten, den transparenten Austausch von Informationen zwischen verschiedenen Werkzeugen oder auch zwischen Entwicklern ermöglichen, sich wiederholende Arbeitsschritte automatisieren und den „gesamten“ Produktentwicklungsprozeß unterstützend steuern. Eine solche Umgebung muß u.a. die folgenden Eigenschaften besitzen [41]:

- **Offenheit:** Diese Eigenschaft bezieht sich auf die Fähigkeit, „beliebige“ Werkzeuge innerhalb der Umgebung einzubeziehen bzw. auszutauschen.
- **Flexibilität:** Diese Eigenschaft bedeutet die Fähigkeit der Anpassung der Umgebung an unternehmens- bzw. aufgabenspezifische Gegebenheiten.

An dieser Stelle sei betont, daß die eigentliche Entwurfsumgebung nur eine Menge von *Basisfunktionalitäten* bereitstellt, die es ermöglichen, die spezifischen Funktionalitäten der vorhandenen bzw. einzubeziehenden Softwarewerkzeuge innerhalb der Gesamtumgebung flexibler einzusetzen.

Die Realisierung einer Entwurfsumgebung zur Unterstützung von Entwurfsprozessen ist zunächst Gegenstand der weiteren Betrachtungen. Daran anschließend wird ein Framework für das Prozeßmanagement innerhalb einer Entwurfsumgebung angegeben, mit welchem die auszuführenden Entwurfsprozesse explizit betrachtet werden können. Der Beitrag schließt mit der Darstellung eines möglichen Anwendungsszenarios und der dafür notwendigen Prototypumgebung.

2 Realisierungsstrategien für Entwurfsumgebungen

Für den Entwurf und die Realisierung einer integrierten, unternehmensweiten Entwurfsumgebung ist sowohl die Betrachtung der Entwurfsdokumente und deren Zuordnung zu einer konkreten Produktstruktur als auch der Entwurfsprozesse zwingend erforderlich. In der Literatur wurden Realisierungsansätze für Entwurfsumgebungen vorgeschlagen, die unterschiedliche Aspekte betonen. Dazu gehören der *Entwurfswerkzeug-*, der *Entwurfsprozeß-* und der *Entwurfsdokumentaspekt*. Diese Aspekte werden in den jeweiligen Ansätzen in den Mittelpunkt der Betrachtungen gestellt. Im folgenden sollen einige Methoden und Techniken diskutiert werden, die eine Realisierung einer Entwurfsumgebung unter dem jeweiligen Aspekt betrachten.

2.1 Entwurfswerkzeugaspekt

In den Unternehmen ist eine Vielzahl unterschiedlicher Werkzeuge zur Unterstützung von Entwurfsprozessen vorhanden. Für eine Entwurfsumgebung mit Betonung des Entwurfswerkzeugaspekts ist daher die Bereitstellung von entsprechenden Methoden und Modellen erforderlich, die eine Verwendung dieser Werkzeuge innerhalb der neuen Umgebung ermöglichen. Dabei werden die Entwurfswerkzeuge als wiederver-

wendbare Softwarebausteine betrachtet, die in einer „beliebigen“ Form zu neuen Entwurfswerkzeugen innerhalb einer Entwurfsumgebung kombiniert werden können. Aus den möglichen Ansätzen zur Wiederverwendung von Softwarewerkzeugen bzw. -modulen (vgl. beispielsweise [27]) bietet sich nur der Ansatz der *Komposition von Komponenten* an. Weder der Zugriff auf die interne Struktur der vorhandenen Entwurfswerkzeuge, noch die Realisierung eines Universalsystems für die Produktentwicklung, welches dann auf ein konkretes Unternehmen adaptiert werden kann, sind praktikabel.

Der Ansatz der Komposition von Komponenten zur Realisierung einer Entwurfsumgebung geht von einer generellen komponentenbasierten Sichtweise aus, wobei alle Entwurfswerkzeuge als spezielle Komponenten der Umgebung betrachtet werden. Eine *Komponente* ist ein wiederverwendbarer Softwarebaustein, der eine Black-Box darstellt, die die interne Realisierung der komponentenspezifischen Funktionalitäten verbirgt und auf die nur über bekannte (standardisierbare) Schnittstellen zugegriffen werden kann. Darüber hinaus muß eine Komponente Schnittstellen besitzen, um eine Zusammenarbeit mit anderen Komponenten zu ermöglichen [3]. Die Verwendung des Komponentenbegriffs zur Abstraktion von den konkreten Softwarewerkzeugen ist die Voraussetzung, um eine explizite Verbindung und Abstimmung zwischen Werkzeugen zu einem funktionierenden Gesamtsystem zu realisieren. Dieses wird im allgemeinen mit *Integration* und in einer Entwurfsumgebung mit *Komponentenintegration* bezeichnet [43].

In Anlehnung an [49] kann eine Integration von Komponenten auf vier unterschiedlichen Ebenen erfolgen: auf der *Daten-*, der *Steuerungs-*, der *Prozeß-* und der *Präsentationsebene*. Auf diesen Ebenen ist eine separate Verbindung von Komponenten durch eine Bereitstellung geeigneter Modelle möglich. So steht beispielsweise mit dem *Integrierten Produktmodell* ein entsprechendes Modell zur Integration von Komponenten auf der Datenebene zur Verfügung [21]. Innerhalb einer Entwurfsumgebung ist eine separate Betrachtung einzelner Ebenen nicht ausreichend, sondern es sind vielmehr alle Ebenen gleichwertig zu berücksichtigen. In der Literatur wurde dazu eine Reihe von *Mechanismen zur Komponentenintegration* eingeführt. Diese Mechanismen beziehen sich in der Regel auf die Bereitstellung von Methoden und Techniken für die Verknüpfung von Komponentenschnittstellen. Zur Verbindung von Komponenten über ihre im allgemeinen heterogenen Schnittstellen ist eine einheitliche Interpretation dieser Schnittstellen erforderlich. Ein geeignetes Hilfsmittel dazu stellen *Schnittstellenbeschreibungssprachen* dar, mit denen eine abstrakte Beschreibung aller Operationen und Datentypen unterstützt wird [45].

Geeignete Modelle zur Verbindung von Komponenten können u.a. grob in die Klassen Komponententechnologien, Kompositions- und Koordinationsmodelle und Konfigurationssprachen unterteilt werden. *Komponententechnologien*, wie die *Common Object Request Broker Architecture* (CORBA) der Object Management Group [32] oder das *Distributed Component Object Model* (DCOM) von Microsoft [39], sind Ansätze zur Realisierung der Komponentenintegration in heterogenen Umgebungen. Ein direkter Zugriff auf die jeweiligen Quelltexte der einzelnen Komponenten ist dabei nicht notwendig. Im Gegensatz dazu stehen die *Kompositions- und Koordinationsmodelle*, wie *Formale Konnektoren* [4], bei denen nicht nur die Komponenten betrachtet werden, sondern darüber hinaus auch die Beziehungen zwischen den zu

integrierenden Komponenten. Das generelle Ziel von *Konfigurationssprachen*, beispielsweise *Darwin* [28], ist die Bereitstellung von Methoden und Konzepten für eine vollständige Architekturbeschreibung eines Systems. Dazu werden die Kompositions- und Koordinationsmodelle um die Möglichkeit der Spezifikation der Gesamtarchitektur eines Systems erweitert.

2.2 Entwurfsprozeßaspekt

Die Realisierung von komplexen Produkten erfordert die Ausführung von unterschiedlichen Entwurfsprozessen in einer bestimmten Reihenfolge. In einer Entwurfsumgebung mit Betonung des Entwurfsprozeßaspekts bilden diese Prozesse den Mittelpunkt der Betrachtungen. Die Ausführung von Prozessen erfordert in der Umgebung die Bereitstellung entsprechender Methoden und Techniken für das Prozeßmanagement. In der Literatur wurden rechnerunterstützte Umgebungen zur Unterstützung der Ausführung spezieller Aufgaben (z.B. Entwurfsaufgaben) entwickelt, welche den Anspruch erheben, die Ausführung von Entwicklungsprozessen hinsichtlich aller Aspekte, beispielsweise der einheitlichen Bereitstellung von Entwurfsdokumenten und Komponenten, zu unterstützen. Wichtige Vertreter dieses Ansatzes stellen die *Groupware- (Workflow-) Umgebungen* [9] und die *Softwareentwicklungsumgebungen* [19, 36] dar. Die generelle Zielstellung dieser Umgebungen liegt in der Bereitstellung von Methoden und Konzepten zur rechnerunterstützten Entwicklung spezifischer Produkte, wobei der auszuführende Prozeß innerhalb der jeweiligen Umgebung direkt spezifiziert und ausgeführt werden kann. Von diesen Umgebungen wird im Sinne der Bereitstellung von realisierungsspezifischen Funktionalitäten nichts „produziert“, sondern sie dienen lediglich zur Unterstützung der Ausführung von Entwicklungsprozessen. Im allgemeinen werden diese Umgebungen auch als *prozeßzentrierte Entwicklungsumgebungen* (Process-centered Engineering Environment - PCEE) bezeichnet.

Die PCEE stellen eine neue Klasse von Softwaresystemen dar, in der die Benutzer organisatorisch in den Entwicklungsprozeß integriert werden. Dazu werden von der Umgebung die entsprechenden Technologien bereitgestellt. Dies sind sowohl Basistechnologien, wie Funktionalitäten zur Spezifikation von Prozessen, als auch Anwendungsfunktionalitäten, die zuvor mit Hilfe entsprechender Basistechnologien in die PCEE integriert wurden. Eine PCEE unterstützt einen Benutzer bei der Ausführung seiner Aufgaben insofern, daß diese Vorgaben bezüglich der Reihenfolge der Ausführungsschritte angibt, d.h. die Umgebung „steuert“, wie die Arbeit auszuführen ist und nicht, was zu realisieren ist. Allgemein führt die Einführung von PCEE zu einer Verbesserung der Qualität der erzeugten Produkte und der Produktivität der daran beteiligten Benutzer [11]. Zur Realisierung einer Entwurfsumgebung stellen die PCEE eine geeignete Ausgangsbasis dar. Gründe dafür liegen vor allem in der generellen Unterstützung der auszuführenden Entwurfsprozesse und in der einheitlichen Verwaltung von Entwurfsdokumenten. Aufgrund der speziellen Eigenschaften dieser Prozesse gestaltet sich die direkte Anwendung der Methoden und Techniken zur Prozeßunterstützung im Engineeringbereich jedoch schwierig. So werden beispielsweise spezifische Prozeßmodellierungsmethoden innerhalb spezieller PCEE verwen-

det, deren Einsatz im Engineeringbereich problematisch ist. Daher können diese nur bedingt angewendet werden.

2.3 Entwurfsdokumentaspekt

In einer Entwurfsumgebung auf der Basis der Betonung des Entwurfsdokumentaspekts werden die Produktstruktur bzw. die zu erzeugenden Entwurfsdokumente eines Produktes in den Mittelpunkt der Betrachtungen gestellt. Die Realisierung eines komplexen Produktes beinhaltet die Erzeugung, Weiterverarbeitung und Speicherung einer Vielzahl unterschiedlicher Entwurfsdokumente, wobei diese in der Regel bezüglich der Struktur eines Produktes verwaltet werden. Vor dem Hintergrund der Globalisierung der Märkte und der daraus resultierenden kürzeren Produktentwicklungs- und Lieferzeiten gewinnt die *Produktdatenverwaltung* in Hinblick auf eine effiziente Verwaltung dieser Mengen an Entwurfsdokumenten zunehmend an Bedeutung [14]. Dafür ist die Bereitstellung geeigneter Methoden und Konzepte zur Speicherung und vor allem jedoch für einen effizienten Zugriff auf diese Dokumente innerhalb einer Entwurfsumgebung erforderlich.

Damit eine rechnerinterne Verarbeitung von Produkten einschließlich der Produktstruktur ermöglicht werden kann, ist die Angabe eines *Produktmodells* notwendig [47]. Die Definition eines Produktmodells hängt im allgemeinen von dem jeweiligen betrachteten Anwendungskontext ab. Die generelle Zielsetzung dieses Modells liegt in der Unterstützung einer durchgängigen Informationsverarbeitung in allen Phasen des Produktlebenszyklus. In der Literatur wurden unterschiedliche Ansätze zur Realisierung eines Produktmodells vorgeschlagen [37], wobei das *Integrierte Produktmodell* [21] den bedeutensten Ansatz darstellt. Die Definition eines integrierten Produktmodells ist eine interdisziplinäre Aufgabe, für die eine umfassende Spezifikationsmethodik erforderlich ist. So wurde für die technischen Anwendungsbereiche des Maschinen- und Anlagenbaus innerhalb des ISO-Standards 10303 der *Standard for the Exchange of Product Model Data* (STEP) angegeben [34], mit dem die Spezifikation eines integrierten Produktmodells möglich ist. Mit STEP steht ein Standard zur Verfügung, mit dem über den gesamten Lebenszyklus eines Produktes die Spezifikation der Produktinformationen in einem einheitlichen und vor allem anwendungsunabhängigen Format möglich ist.

Rechnerunterstützte Umgebungen auf der Basis des Entwurfsdokumentaspekts werden heute beispielsweise durch EDM-Systeme bereitgestellt. Diese können direkt in Entwurfsumgebungen eingesetzt werden.

2.4 Lösungsansatz

Die Realisierung einer Entwurfsumgebung zur Unterstützung von Entwurfsprozessen auf der Basis der Betonung einzelner Aspekte ist aus heutiger Sicht nicht mehr ausreichend. So werden beispielsweise von aktuellen EDM-Systemen „nur“ Funktionalitäten für die Produktdatenverwaltung, aber keine ausreichende Möglichkeiten zur Modellierung und Ausführung von Entwurfsprozessen bereitgestellt [2, 14]. Eine

diesbezügliche Unterstützung ist aber gerade im Bereich der verteilten (unternehmensweiten bzw. unternehmensübergreifenden) Prozeßausführung notwendig [22]. Die Verwendung einer prozeßzentrierten Entwurfsumgebung beinhaltet für die Unternehmen aufgrund der expliziten Betrachtung auszuführender Entwurfsprozesse eine Reihe von Vorteilen. Dazu gehören u.a.:

- In einer PCEE ist es prinzipiell möglich, Ausführungsreihenfolgen von Entwurfsprozessen anzugeben, mit deren Ausführung ein bestimmtes Ziel erreicht werden soll. Damit wird die Realisierung eines Produktes und vor allem die dazu erforderlichen Entwurfsprozesse von Anfang an verwaltet. Dabei werden nur Prozesse betrachtet, die diesbezügliche Vorgaben ermöglichen bzw. beispielsweise aufgrund von gesetzlichen Bestimmungen erfordern.
- Die zur Ausführung von Entwurfsprozessen erforderlichen Entwurfswerkzeuge und -dokumente können in einer PCEE einheitlich hinterlegt und dann unter Umständen automatisch durch die Umgebung aufgerufen bzw. bereitgestellt werden. Damit werden die Benutzer einer Entwurfsumgebung bei der Auswahl der notwendigen Werkzeuge und Dokumente unterstützt.
- Entwurfsprozesse, die für ihre Ausführung keine Benutzerinteraktion benötigen, können durch die PCEE automatisch ausgeführt werden.
- Ist zur Erstellung von Entwurfsdokumenten eine Kooperation zwischen verschiedenen Entwicklungsingenieuren notwendig, kann dies von der Umgebung durch Bereitstellung entsprechender Werkzeuge, mit denen beispielsweise *Concurrent* oder *Simultaneous Engineering* [18, 48] möglich ist, unterstützt werden.
- Um Entwurfsprozesse nach ihrer Ausführung auf Optimierungsmöglichkeiten bezüglich einer Verbesserung der Prozeßausführung hin untersuchen zu können, ist die Speicherung aller relevanten Ausführungsdaten erforderlich. Dies kann durch die Umgebung automatisch erfolgen.

Die Autoren schlagen eine Kombination von Ansätzen der Abschnitte 2.1-2.3 zu einer komponentenbasierten Entwurfsumgebung vor, wobei die auszuführenden Entwurfsprozesse in den Mittelpunkt der Betrachtungen gestellt werden sollen. Diese Entwurfsumgebung soll im folgenden als *Prozeßzentrierte Produktentwicklungsumgebung* (Process-centered Product Engineering Environment - PPEE) bezeichnet werden. Damit die Umgebung an konkrete Unternehmensbedürfnisse angepaßt werden kann, wird diese generell als *Framework* [38] realisiert. Zu diesem Framework gehört u.a. ein Modell zur Integration von Komponenten [41]. Dieses kann der Klasse der Kompositions- und Koordinationsmodelle zugeordnet werden und ermöglicht die explizite Verbindung von heterogenen Komponenten. Für die strukturierte Speicherung von Entwurfsdokumenten zu Produkten sollen die Konzepte von EDM-Systemen angewendet werden. Diese werden mit Hilfe der Komponentenintegration in die Umgebung „einbezogen“. Für die explizite Unterstützung von Entwurfsprozessen soll ein entsprechendes Modell eingesetzt werden, welches auf den in Abschnitt 2.2 erläuterten Methoden und Techniken aufbaut, wobei entsprechende Erweiterungen bzw. Ergänzungen bezüglich der Eigenschaften des Engineeringbereichs erforderlich sind.

Innerhalb des Frameworks werden bekannte (vorhandene) Techniken bezüglich der Komponentenintegration und der Produktdatenspeicherung verwendet. Aus diesem Grund stellt ein Modell, mit welchem die explizite Unterstützung von Entwurfsprozessen innerhalb einer PPEE unterstützt werden kann, den Gegenstand der weiteren Betrachtungen dar. Dieses Modell ist die Grundlage eines spezifischen Frameworks zur Realisierung der Unterstützung von Entwurfsprozessen.

3 Framework für das Prozeßmanagement

Traditionell wird die Ausführung von Entwurfsprozessen durch eine Reihe von Entwurfswerkzeugen unterstützt. So wird beispielsweise der Prozeß der „Konstruktion eines Einzelteils“ mit Hilfe eines CAD-Werkzeugs ausgeführt. Sollen Informationen oder Funktionalitäten zwischen verschiedenen Werkzeugen ausgetauscht werden, so wird dieses in der Regel über spezifische Schnittstellen zwischen den Werkzeugen realisiert. Hinzu kommt noch die Tatsache, daß die Unternehmen aus Kostengründen die Fertigungstiefen ihrer Produkte reduzieren und Teilaufträge an andere Unternehmen vergeben. Für die Entwicklungsprozesse bedeutet dies, daß nicht länger nur die internen Unternehmensstrukturen bzw. -abläufe zu betrachten sind, sondern das zunehmend auch die Kunden und Lieferanten eines Unternehmens mit in die Betrachtungen einzubeziehen sind.

Der Einsatz von systemübergreifenden Entwurfsumgebungen ermöglicht den Unternehmen nicht nur eine effektivere Anwendung von Entwurfswerkzeugen und eine einheitliche Verwaltung der Entwurfsdokumente in allen Phasen der Produktentwicklung, sondern auch eine explizite Unterstützung bei allen auszuführenden Entwurfsprozessen. Die Notwendigkeit einer expliziten Unterstützung von Prozessen wurde zunächst für den Bereich der Softwareentwicklung erkannt [10], in welchem Softwareingenieure mit Werkzeugen einem vorgegebenen Prozeß entsprechend interagieren, um neue Softwareprodukte zu erzeugen. Nach [12] können ähnliche Konzepte und Methoden auch in anderen Bereichen beispielsweise im Bereich des Business-Engineerings eingesetzt werden. Durch eine Bereitstellung eines entsprechenden Frameworks zur Unterstützung von Prozessen, das prozeßspezifische Konzepte und Techniken beinhaltet und auf spezifische Bereiche angepaßt werden kann, ist dieses möglich. Dabei ist zu beachten, daß nicht nur die Entwurfsprozesse, sondern auch die angrenzenden Bereiche, wie die unterstützenden Werkzeuge oder die zu erzeugenden Dokumente, mit in die Betrachtungen einbezogen werden müssen. Auch Aspekte aus dem *Computer Supported Cooperative Work* Bereich, wie die Koordinierung von Personen über einen bestimmten Zeitraum oder über bestimmte geographische Gegebenheiten, sind zu berücksichtigen [24].

Für die Realisierung des Frameworks zur Unterstützung von Entwurfsprozessen wird ein komponentenbasierter Ansatz vorgeschlagen, um auf die Erfordernisse im Rahmen der sich ändernden Anforderungen an eine generelle PPEE geeignet reagieren zu können [17]. So kann beispielsweise das erfolgreiche Abschließen eines Projektes die auftragsgebundene Integration von speziellen Werkzeugen in die Umgebung oder die Verwendung von speziellen Prozessen erfordern. Auch die generellen Anforderungen an eine Entwurfsumgebung bezüglich der Offenheit und Flexibilität

können mittels eines komponentenbasierten Ansatzes erfüllt werden. Prinzipiell ist dazu die Verwendung von Techniken der PCEE möglich. Problematisch wirkt sich dabei allerdings aus, daß diese Konzepte für andere Bereiche, wie die Unterstützung von Softwareentwicklungsprozessen, entwickelt wurden, so daß eine direkte Anwendung innerhalb einer PPEE aufgrund der Eigenschaften des Engineeringbereichs selten möglich ist. So werden in diesen Umgebungen beispielsweise die zu erzeugenden Dokumente nicht anhand einer allgemeinen (Software-)Produktstruktur verwaltet.

Um Entwurfsprozesse innerhalb einer PPEE unterstützen zu können, soll im folgenden der Ansatz für ein Framework dargestellt werden, welcher sich von anderen Ansätzen bezüglich der Prozeßunterstützung unterscheidet. Als generelle Herangehensweise wurde ein *Partialmodellarchitekturansatz* [44] gewählt. Ein solcher Ansatz hat sich beispielsweise im Bereich der Produktdatenmodelle bereits bewährt [35]. Innerhalb eines Partialmodells werden dabei die zur Lösung einer speziellen Aufgabe notwendigen Datenmodelle zusammengefaßt. Die Partialmodelle zur Unterstützung des Prozeßaspekts bilden im weiteren die Grundlage von unabhängigen (Teil-)Komponenten, die im Rahmen der Prozeßunterstützung benötigt werden. Zu diesen Komponenten gehören im einzelnen:

- **Prozeßkomponente:** Innerhalb dieser Komponente werden alle Funktionalitäten bezüglich der Spezifikation, Instantiierung und Ausführung von Entwurfsprozessen zusammengefaßt. Darüber hinaus gehören dazu auch Funktionalitäten zur Überwachung der Ausführung sowie zur Validierung und Verifikation von Prozessen.
- **Zustandskomponente:** Innerhalb von EDM-Systemen werden die Lebenszyklen der Entwurfsdokumente häufig mit Hilfe von Zustandsdiagrammen beschrieben. Dieses Konzept soll außerdem zur Beschreibung der Zustände der Ausführung von Prozessen eingesetzt werden. Die Komponente stellt die dazu notwendigen Funktionalitäten bereit.
- **Zugriffskomponente:** Im Rahmen der unternehmensweiten Ausführung von Prozessen sind u.a. Funktionalitäten zur Realisierung des Zugriffs auf Prozesse erforderlich. Dies ist besonders im Kunden-Lieferanten-Bereich der verteilten Ausführung von Prozessen notwendig, wo einzelne Teilprozesse „außerhalb“ eines Unternehmens ausgeführt werden. Die Komponente stellt die entsprechenden Funktionalitäten bereit, wobei ein *Rollenkonzept* zum Einsatz kommt.
- **Entwurfsdokumentekomponente:** Zur Ausführung von Prozessen ist die Bereitstellung bzw. Verwaltung der erforderlichen Entwurfsdokumente notwendig. Dazu sind neben einer Verwaltung der Dokumente, die auf der Basis eines Produktstrukturmodells erfolgt, auch entsprechende *CheckIn-* und *CheckOut-*Mechanismen für die Dokumente anzubieten [15]. Die dazu erforderlichen Funktionalitäten werden durch diese Komponente bereitgestellt.

Den Mittelpunkt des Ansatzes bildet die Prozeßkomponente, die alle notwendigen Funktionalitäten zur Modellierung und Ausführung von Entwurfsprozessen bereitstellt (vgl. Abbildung 1). Die zu spezifizierenden Prozeßmodelle beinhalten daher nicht nur Informationen über die einzuhaltenden Ausführungsbeziehungen der einzelnen (Teil-)Prozesse, sondern auch Informationen, die für die Ausführung von

atomaren Prozessen erforderlich sind. Im Bereich der Entwurfsprozesse gehören dazu neben den für die Ausführung notwendigen Entwurfsdokumenten und Rollen, auch Informationen über die Komponenten bzw. die Schnittstellen von Komponenten mit denen die Ausführung der Entwurfsprozesse unterstützt werden kann.

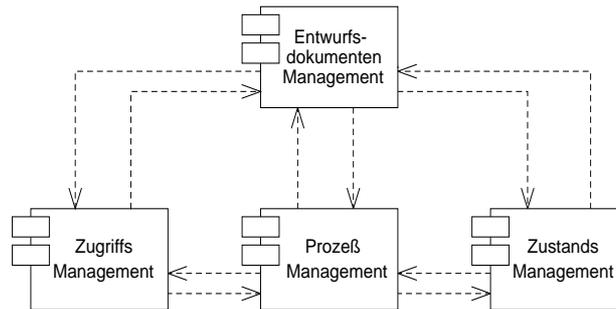


Abbildung 1: Komponenten einer PEE in Anlehnung an UML [8]

Innerhalb der Prozeßkomponente wird wiederum ein komponentenbasierter Ansatz angewendet, um eine strikte Trennung zwischen den eigentlich unabhängigen Bestandteilen eines Prozesses realisieren zu können. Dazu gehören eine *Prozeßmodellierungs-*, eine *Prozeßausführungs-*, und eine *Kontrollflußkomponente*. Diese Trennung wurde mit dem Ziel eingeführt, auf der einen Seite eine einheitliche Schnittstelle zur Modellierung von Prozeßabläufen und auf der anderen Seite davon getrennte Ausführungskomponenten für Prozesse und den Kontrollfluß zwischen diesen Prozessen zu erhalten. Aufbauend auf diese Trennung ist es möglich, domänenspezifische Modellierungsmethoden (z.B. *Funsoft-Netze* [23] für Geschäftsprozesse), unternehmensspezifische Ausführungsmethoden (zum Beispiel auf spezielle Hardware angepaßt) und „standardisierte“ Kontrollflußmethoden (zum Beispiel auf der Basis von Petrinetzen) einzusetzen. Im einzelnen beinhalten diese Komponenten folgende Funktionalitäten:

- **Prozeßmodellierungskomponente:** Mit Hilfe der von dieser Komponente bereitgestellten Funktionalitäten ist es möglich, Prozesse und Prozeßabläufe zu spezifizieren. Zusätzlich zu den reinen Spezifikationsmethodiken wird noch eine Reihe weiterer Funktionalitäten bereitgestellt. Dazu gehören beispielweise Funktionalitäten zur Präsentation von speziellen Prozeßausführungszuständen oder auch zur Analyse und Verifikation entsprechender Prozeßmodelle. Die Funktionalitäten dieser Komponenten hängen im wesentlichen von der konkret anzuwendenden Prozeßmodellierungsmethode ab.
- **Prozeßausführungskomponente:** Die im Rahmen der Spezifikation von Prozessen festgelegten spezifischen Ausführungsinformationen, beispielsweise die bereitzustellenden Entwurfsdokumente oder Rollen, bilden die Grundlage der Funktionalitäten dieser Komponente. Auf der Basis dieser Informationen werden durch die Komponente Funktionalitäten zum Instantiieren, Ausführen und Überwachen von Prozessen bereitgestellt. Für die eigentliche Ausführung der Prozesse

wird ein zustandsbasierter Ansatz verwendet, bei dem die Ausführung indirekt über die dabei auftretenden Ausführungszustände beschrieben wird.

- **Kontrollflußkomponente:** Die Aufeinanderfolge von Teil-Prozessen innerhalb komplexer (d.h. zusammengesetzter) Prozesse wird in der Regel mit Hilfe des sogenannten *Kontrollflusses* ausgedrückt. Ein Kontrollfluß entspricht dabei der Vorgabe der Ausführungsreihenfolge von (atomaren) Prozessen entsprechend einer definierten Ablauflogik mit sequentiellen, parallelen, alternativen und zusammenführenden Strukturen. Im Rahmen dieser Komponente werden Funktionalitäten bereitgestellt, mit denen der Kontrollfluß von komplexen Prozessen spezifiziert und ausgeführt werden kann, wobei einfache *Stellen/Transitionsnetze* [6] verwendet werden.

Um die Funktionalitäten der Prozeßkomponente innerhalb einer PPEE bereitstellen zu können, ist eine Integration aller drei Teil-Komponenten erforderlich. Auf dieser Basis können dann spezifische Prozeßmodelle mit der Modellierungskomponente spezifiziert und mit der Prozeßausführungs- bzw. Kontrollflußkomponente einheitlich ausgeführt werden. Da die Sub-Komponenten auf der Basis von komponentenspezifischen Datenmodellen realisiert werden, erfordert die Integration der Komponenten prinzipiell deren Abstimmung bzw. Abbildung aufeinander. Zur eigentlichen Abstimmung der Datenmodelle ist es daher denkbar, spezifische Abbildungen zwischen den einzelnen Modellen anzugeben. Aufgrund der Verwendung eines frameworkbasierten Realisierungsansatzes für die PPEE und der damit verbundenen Idee der Unabhängigkeit des Lösungsansatzes von konkreten Realisierungsdetails, können keine spezifischen Abbildungen angegeben werden, da dies u.a. die Vorgabe einer spezifischen Prozeßmodellierungsmethode mit den zugehörigen Abbildungsinformationen beinhaltet. Prinzipiell ist außerdem die Verwendung „beliebiger“ Prozeßmodellierungsmethoden denkbar bzw. notwendig. So kann beispielsweise innerhalb einer, auf ein konkretes Unternehmen instantiierten PPEE die Spezifikation und Ausführung von Prozeßmodellen mit Hilfe von *Ereignisgesteuerten Prozeßketten* [46] und von *Funsoft-Netzen* notwendig sein.

Laut [7] wird gefordert, domänenspezifische Methoden und Werkzeuge zur Modellierung von Prozessen einzusetzen. Nur dadurch kann ein Benutzer während der Prozeßausführung geeignet unterstützt werden. Das Verständnis für die Prozesse wird durch die Auswahl von Spezifikationsmethodiken, die an das jeweilige Anwendungsumfeld der Benutzer angepaßt sind, erhöht. Darüber hinaus ist für die rechnerunterstützte Ausführung von Prozessen ein formales Prozeßmodell erforderlich.

Um die Unabhängigkeit des Lösungsansatzes für die PPEE von konkreten Prozeßmodellierungsmethoden zu gewährleisten, wird die Einführung eines *generischen Prozeßmodells* als Datenmodell für die Prozeßmodellierungskomponente vorgeschlagen [16]. Mit Hilfe dieses Modells steht innerhalb der Prozeßmodellierungskomponente ein Datenmodell zur Verfügung, mit dem die Abbildung zwischen den Komponenten unabhängig von einer konkreten Modellierungsmethode möglich ist. Die spezifizierten Prozeßmodelle werden im Rahmen dieses Ansatzes mit Hilfe einer speziellen Transformation in das generische Prozeßmodell abgebildet. In Abbildung 2 ist der gewählte Lösungsansatz abschließend angegeben.

Für die Realisierung der Prozeßkomponente ist die Verwendung von Workflow Management Systemen (WfMS) nur bedingt geeignet, da deren Einsatz nur für Routinetätigkeiten sinnvoll ist, wobei die Definition der Arbeitsabläufe vor der eigentlichen Durchführung erforderlich ist [29]. Außerdem müssen zur Realisierung einer prozeßzentrierten Herangehensweise mittels WfMS die eigentlichen Prozesse klar definierbar und eine mögliche Aufeinanderfolge nach genau vorgegebenen Regeln gegeben sein. Zusätzlich müssen allen beteiligten Kommunikationspartnern eindeutig Funktionen, Rollen und Kompetenzen zugeordnet werden können und die im Arbeitsprozeß entstehenden bzw. zu verwendenden Dokumente standardisierbar sein. Darüber hinaus ist in vielen WfMS die explizite Modellierung von Zuständen eines Workflows nicht möglich. Dadurch können Trigger und externe „Auswahlen“ nicht richtig verwaltet werden [1]. Entwurfsprozesse sind aber durch eine ständige Verzahnung der Planungs- und Ausführungsschritte gekennzeichnet. Weiterhin sind diese Prozesse im Vorfeld nur grob vorhersagbar und müssen zur Laufzeit ständig an die jeweiligen neuen Gegebenheiten angepaßt werden. Außerdem kann die weitere Planung von bereits ausgeführten Arbeitsschritten abhängen. Die zuvor genannten Voraussetzungen werden von derzeitigen kommerziell verfügbaren WfMS daher nicht oder nur bedingt erfüllt, wodurch der Einsatz im Rahmen einer Entwurfsumgebung und damit für die Prozeßkomponente auch nur bedingt möglich ist.

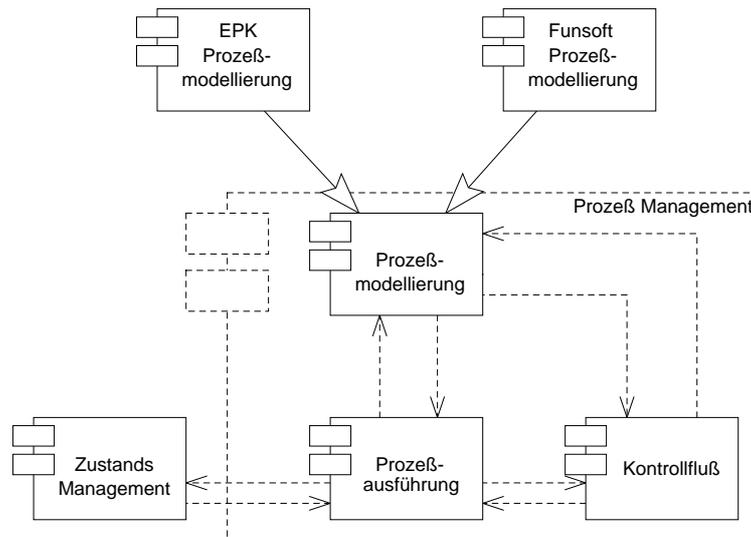


Abbildung 2: Teil-Komponenten der Prozeßkomponente

Mit Hilfe des vorgeschlagenen Frameworks zur Unterstützung von Prozessen können innerhalb einer PPEE Entwurfsprozesse spezifiziert und ausgeführt werden, wobei nur Prozesse betrachtet werden sollen, deren rechentechnische Unterstützung möglich bzw. sinnvoll ist. Für die Einführung bzw. Anpassung des Frameworks für eine PPEE an ein konkretes Unternehmen wird dabei ein *Vorgehensmodell* verwendet, in dem grob zwischen einer *Analyse-*, *Spezifikations-* und *Realisierungsphase* unterschieden

wird. Darüber hinaus wird in der Spezifikationsphase zusätzlich bezüglich der Spezifikation von Produktinformationen, Werkzeugen und Prozessen unterschieden. Im folgenden wird ein mögliches Anwendungsszenario für die vorgeschlagene prozeßzentrierte Produktentwicklungsumgebung angegeben.

4 Anwendungsbeispiel

Das Institut für Fertigungstechnik und Qualitätssicherung der Otto-von-Guericke-Universität Magdeburg beschäftigt sich in Zusammenarbeit mit dem Institut für Technische und Betriebliche Informationssysteme u.a. mit Fragen der Entwicklung und des Einsatzes von Informationssystemen zur Unterstützung der Produktentwicklung. Im speziellen wurden in der Vergangenheit umfangreiche Untersuchungen durchgeführt, die sich beispielhaft mit der *Angebotsvorbereitung in Gießereien* befassen. Diese Untersuchungen haben verdeutlicht, daß gerade dieser Bereich bisher nur ungenügend durch die Informationstechnik unterstützt wird. Im Rahmen dieses Abschnittes sollen Aspekte bezüglich der Einführung einer PPEE in das Umfeld der Angebotsvorbereitung eingeführt werden. Dabei beschränken sich die Darstellungen aufgrund des Standes der Entwicklung zunächst auf die Analysephase und die dabei angewendeten Werkzeuge. Der Abschnitt schließt mit der Darstellung einer möglichen Prototypumgebung.

4.1 Anwendungsszenario

Im Rahmen der Realisierung von komplexen Produkten nimmt die Arbeitsvorbereitung eine besondere Stellung ein, da diese durch eine sehr hohe Kostenverantwortung gekennzeichnet ist. Die Entwurfsprozesse der Angebotsbearbeitung umfassen dabei eine Vielzahl von verschiedenen Teilprozessen, die in einer zuvor bestimmten Reihenfolge auszuführen sind. Der Einsatz einer PPEE zur Unterstützung kann die Ausführung dieser Prozesse positiv beeinflussen. Entsprechend dem Vorgehensmodell zur Einführung einer PPEE in ein konkretes Unternehmen ist zunächst eine Analyse der Unternehmenssituation erforderlich. Dazu gehören u.a. die Analyse der auszuführenden Entwurfsprozesse, der daran beteiligten Abteilungen und der unterstützenden Werkzeuge.

Die durchgeführten Analysen in einer Vielzahl von Gießereien haben zur Spezifikation von Entwurfsprozessen für die Angebotsvorbereitung in Gießereien geführt [25, 26]. Generell werden in der Arbeitsvorbereitung zwei Phasen unterschieden. Zum einen die *Phase der Angebotserarbeitung*, die mit dem Eingang einer Anfrage eines Kunden beginnt. An der Erstellung eines wirtschaftlichen Angebots für einen Kunden sind in der Regel verschiedene Abteilungen beteiligt. Zum anderen die *Phase der Erarbeitung von Fertigungsunterlagen*, die jedoch erst nach dem Erhalt eines endgültigen Auftrags ausgeführt wird. Die dabei erstellten Fertigungsunterlagen werden an die Fertigung übergeben. Da jedoch die Qualität der Angebote verbessert werden soll, wird im folgenden nicht der Ist-Zustand der Arbeitsvorbereitung vorgestellt, sondern es wird auf einen theoretischen Idealfall eingegangen. Hierbei werden die

Zeichnungen nicht erst während der Auftragsphase erstellt, sondern bereits in der Angebotsphase. Dieses Vorgehen ist darin begründet, daß das Potential an vorhandenem Expertenwissen so weit wie möglich in der Umgebung rechentechnisch hinterlegt und genutzt werden soll, um die getroffenen Entscheidungen von den Erfahrungen der jeweiligen Mitarbeiter zu abstrahieren.

Aus der Vielzahl von möglichen Prozeßmodellierungsmethoden (vgl. beispielsweise [30]) wurde zur Abbildung des Vorgehens zur Zeichnungs- und Unterlagenerstellung die eher für die Geschäftsprozeßmodellierung geeignete Modellierungsmethode der *Ereignisgesteuerten Prozeßketten* (EPK) verwendet [46]. Diese soll im folgenden kurz eingeführt werden: Prinzipiell bestehen EPK aus einer Reihe unterschiedlicher Modellierungskonstrukte, die zur Beschreibung der verschiedenen Bestandteile eines Prozesses eingesetzt werden können. Dazu gehören u.a. Ereignisse, Funktionen, Konnektoren, Organisationseinheiten und Informationsobjekte. *Ereignisse* werden in Form eines Sechsecks dargestellt und lösen eine oder mehrere Funktionen aus bzw. sind deren Ergebnis. Unter einer *Funktion* werden Handlungen, Tätigkeiten oder Vorgänge verstanden, die beschreiben, was ausgeführt werden soll. Funktionen sind zeitverbrauchende Vorgänge, die in Abhängigkeit von den Vorgangsergebnissen zu unterschiedlichen Ablaufverzweigungen oder auch zu Rücksprüngen führen können. Verzweigungen werden durch *Konnektoren* dargestellt, wobei zwischen den logischen Operatoren UND, ODER und exklusives ODER (XODER) unterschieden wird. Die zeitliche Reihenfolge der auszuführenden Funktionen und damit der erzeugten Ereignisse werden durch den *Kontrollfluß* beschrieben, welcher durch gestrichelte Pfeile dargestellt wird. Weiterhin werden *Organisationseinheiten* (dargestellt durch Ellipsen), beispielsweise die Abteilungen „Vertrieb“ oder „Personalstelle“, mit einer durchgezogenen Linie den Funktionen zugeordnet. Den Funktionen können außerdem *Informationsobjekte*, dargestellt durch Rechtecke, zugewiesen werden. Sie werden durch Pfeile, die den Informationsfluß verdeutlichen sollen, mit den Funktionen verbunden. Um Verfeinerungen darzustellen, werden *Prozeßwegweiser* genutzt, die als abgerundete Rechtecke dargestellt werden und mit einem Pfeil hinterlegt sind.

Die im Rahmen der Untersuchung gewonnenen Ergebnisse zur Arbeitsvorbereitung haben zur Entwicklung von Prozeßabläufen geführt, die ein theoretisches (optimiertes) Vorgehen zur Verbesserung der Arbeitsvorbereitung beschreiben. Beispielhaft ist dazu in Abbildung 3a ein Ausschnitt aus dem Prozeßablauf zur Angebotserarbeitung dargestellt. Dabei wird lediglich die Zeichnungserarbeitung näher betrachtet. Der Ablauf beginnt damit, daß ein angefragtes Gußteil als Neuteil (NT), Ähnlicheil (AT) oder Wiederholteil (WT) klassifiziert wird. Diese Funktion kann erst dann ausgeführt werden, wenn die eingegangenen Unterlagen zur Zeichnungserstellung vorliegen. Die Bearbeitung erfolgt durch den Bereich „Arbeitsvorbereitung“ (AV). Der Funktion werden die Unterlagen (UL) zum angefragten Teil sowie ein Werkzeug „Ähnlicheil-suche zur Klassifikation von Gußstücken“ zugeordnet. Als Ergebnis der Funktion liegen wiederum die Unterlagen vor und die Entscheidung, ob es sich um ein Neuteil, ein Ähnlicheil oder ein Wiederholteil handelt. Um dieses auszudrücken, wurde ein XODER-Konnektor eingefügt.

In Abbildung 3b ist die Verfeinerung der Funktion „Unterlagen für WT erarbeiten“ dargestellt. Der verfeinerte Prozeß beginnt mit der Überprüfung der vorhandenen Unterlagen für das Wiederholteil. Hierbei sind der Funktion die Unterlagen für das

angefragte Teil und die vorhandenen Unterlagen für das Wiederholteil zugeordnet. Die ausführende Organisationseinheit ist wiederum der Bereich „Arbeitsvorbereitung“. Zur Überprüfung der Unterlagen kann an dieser Stelle bereits ein CAD-System, z.B. *SolidWorks* (SW99) der Firma Solidworks Corporation, genutzt werden. Im Ergebnis ist die Entscheidung zu treffen, ob die Zeichnungen in Ordnung sind oder nicht. Treten Probleme bei den Zeichnungen auf, sind im nächsten Schritt die Zeichnungen durch die Konstruktionsabteilung (K) zu ändern bzw. anzupassen. Der jeweilige Bearbeiter nutzt an dieser Stelle ebenfalls ein CAD-System. Sobald die Zeichnungen geändert sind, erfolgt ein Rücksprung zur Funktion „Zeichnungen und UL prüfen“. Sind alle Unterlagen in Ordnung wird der verfeinerte Prozeß verlassen und über den Prozeßwegweiser wieder an den ursprünglichen Ablauf zurückgeben.

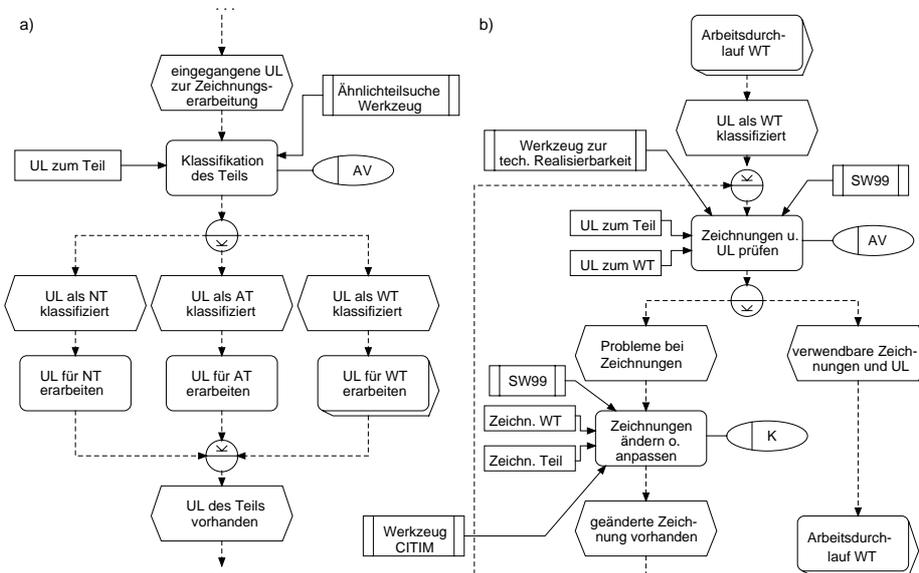


Abbildung 3: Ablauf zur Zeichnung- und Unterlagenerstellung in eEPK [42]

Parallel zur Analyse von Entwurfsprozessen in der Angebotsvorbereitung wurden die Werkzeuge bestimmt, die die Ausführung der einzelnen Prozesse unterstützen. Dazu gehören beispielsweise die datenbankbasierten Werkzeuge zur „Technischen Realisierbarkeit von Gußstücken“, zur „Ähnlichteilssuche“, zur Überprüfung der „Vollständigkeit der Unterlagen“ oder zur „Überprüfung der Fertigungsgerechtigkeit“. Auf eine nähere Darstellung der einzelnen Werkzeuge sei mit dem Verweis auf [5] verzichtet.

4.2 Prototyp

Auf der Basis des in Abschnitt 2 angegebenen Ansatzes zur Bereitstellung einer prozessorientierten Entwurfsumgebung zur Unterstützung der Arbeitsvorbereitung in Gießereien wurde zunächst, unabhängig von einem konkreten Unternehmen, ein

Prototyp der Umgebung realisiert. Dieser Prototyp setzt sich generell aus einem Basissystem und den Komponenten für das Prozeßmanagement, für das Daten-/Dokumentenmanagement und für das Aktivitätenmanagement zusammen. Diese voneinander unabhängigen Komponenten stellen alle spezifischen Funktionalitäten aus dem jeweiligen Anwendungsumfeld bereit. Die Aktivitätenkomponente bietet beispielsweise Funktionalitäten für die Verwaltung von *Aktivitäten* an. Dabei sind Aktivitäten die atomaren Einheiten, die durch Softwaresysteme oder Benutzer ausgeführt werden können. Jede Aktivität besitzt innerhalb der Gesamtumgebung einen Typ, mit dem festgelegt wird, inwieweit die Umgebung deren Ausführung unterstützen kann. Dabei gibt es u.a. einen Typ „ServerActivity“, der ausdrückt, daß die Aktivität ohne Benutzerinteraktion ausgeführt werden kann. Im Gegensatz dazu steht ein Typ „ManualActivity“, der angibt, daß die Aktivität durch einen Benutzer und ohne die Unterstützung durch die Umgebung ausgeführt werden muß. Diese unabhängigen Komponenten werden innerhalb der Entwurfsumgebung zusammengefaßt, damit die für die Umgebung spezifischen Funktionalitäten bereitgestellt werden können. Zusätzlich gehören zu diesem Basissystem auch Funktionalitäten zur Integration von Komponenten, auf die an dieser Stelle aber nicht weiter eingegangen werden soll.

Als Implementierungsplattform des Prototyps wurde, u.a. aufgrund der Anforderung der Integration von heterogenen Softwarewerkzeugen in die Gesamtumgebung, ein auf CORBA basierender Ansatz gewählt. Dazu kam die CORBA-Implementierung ORBacus der Firma Object Oriented Concept Inc. zum Einsatz [33]. Zur Implementierung der Entwurfsumgebung wurde die Programmiersprache JAVA verwendet.

In Abbildung 4 ist der Einsatz der vorgeschlagenen Entwurfsumgebung in einem konkreten Unternehmen beispielhaft angedeutet. Prinzipiell wird die Entwurfsumgebung dabei als *prozeßzentrierte Produktentwicklungsumgebung* (PPEE-Server) innerhalb des Unternehmens an das Unternehmensnetz angeschlossen. Als eine wesentliche Voraussetzung muß angenommen werden, daß jeder Rechnerknoten, der die Funktionalitäten des PPEE-Servers nutzen möchte, ebenfalls temporär oder permanent an das Unternehmensnetzwerk angeschlossen sein muß. Zusätzlich müssen alle Knoten beim PPEE-Server registriert werden. Somit können den Rechnerknoten innerhalb der PPEE eindeutig Funktionen und Kompetenzen zugeordnet werden. Als eine lokale Verbindung zum PPEE-Server muß auf jedem Rechnerknoten ein sogenannter PPEE-Client installiert sein. Dieser Client ist die Schnittstelle des lokalen Rechners zum PPEE-Server, d.h. über diesen Client kann ein lokaler Rechner die angebotenen Funktionalitäten des PPEE-Servers nutzen. Damit sind alle Voraussetzungen geschaffen, um die Funktionalitäten des PPEE-Servers unternehmensweit einsetzen zu können. Ein spezieller Rechnerknoten kann beispielsweise seine bereitgestellten Funktionalitäten¹ über den PPEE-Client beim PPEE-Server anmelden. Dadurch teilt der Client der Gesamtumgebung mit, daß er mit Hilfe seiner lokalen Softwarewerkzeuge diese Aktivität ausführen kann. Durch die Verwendung eines Rollenkonzeptes läßt sich dieses Konzept noch um einen Zugriffsschutz erweitern, so daß nur bestimmte Benutzer bestimmte Aktivitäten ausführen können. Benötigt ein

¹ Diese Funktionalitäten werden innerhalb des PPEE-Servers als Aktivitäten bezeichnet.

Rechnerknoten eine bestimmte Aktivität, dann teilt er dies der Umgebung mit, und die Umgebung „veranlaßt“ die Ausführung der entsprechenden Aktivität durch den Rechnerknoten, der diese Aktivität bereitstellt. Es sei darauf verwiesen, daß es prinzipiell auch möglich ist, komplexe Aktivitäten innerhalb der Gesamtumgebung anzumelden. Dazu wird diese Aktivität durch einen Benutzer mit Hilfe eines speziellen Prozeßablaufes spezifiziert und als *komplexe Aktivität* bei der Umgebung angemeldet. Die dazu notwendigen Funktionalitäten werden von der Entwurfsumgebung, genauer vom Prozeß Management, bereitgestellt.

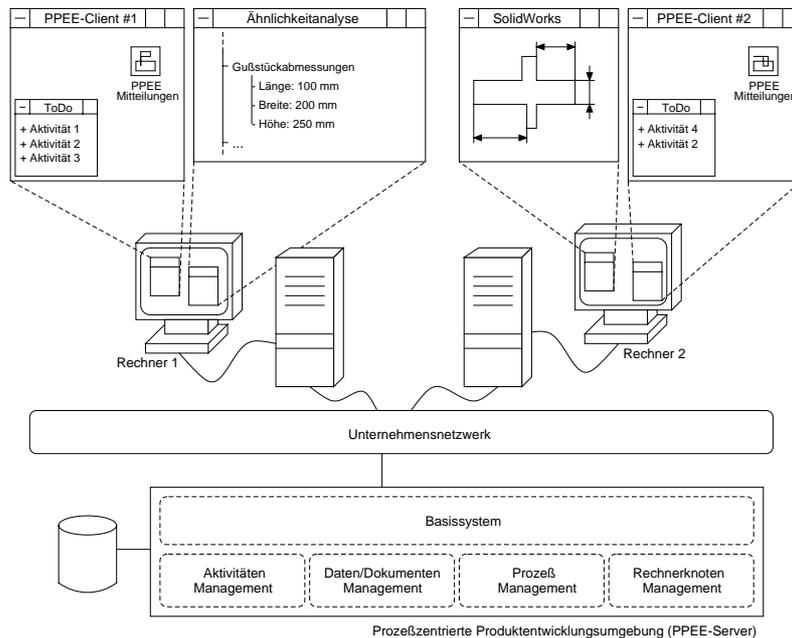


Abbildung 4: Umgebungsarchitektur für eine PPEE

Mit Hilfe dieses Prototyps kann die Arbeitsvorbereitung in Gießereien innerhalb eines Unternehmens entsprechend unterstützt werden. Dazu müssen zunächst alle notwendigen Abteilungen mit dem PPEE-Server verbunden werden. Die von den Abteilungen bereitgestellten Funktionalitäten, die innerhalb der Arbeitsvorbereitung benötigt werden, können dann mit Hilfe des PPEE-Client bei der Umgebung „angemeldet“ werden. Dazu gehört beispielsweise auch der Prozeß „Zeichnungserarbeitung“, welcher mit Hilfe von EPK spezifiziert wurde (vgl. Abbildung 3) und durch den Rechnerknoten der Abteilung „Arbeitsvorbereitung“ bereitgestellt wird. Der PPEE-Server beinhaltet Funktionalitäten, mit denen dieser Prozeß zunächst in das generische Prozeßmodell transformiert und dann durch die Umgebung den anderen Abteilungen zur Ausführung bereitgestellt wird. Auf dieser Basis kann dann beispielsweise der Prozeß „Erstellung eines Angebots“ durch verschiedene Abteilungen mit Unterstützung des PPEE-Servers unternehmensweit ausgeführt werden.

Mit diesem Prototyp steht prinzipiell ein allgemeines Framework bereit, mit dem die unternehmensweite Ausführung von Entwurfsprozessen unterstützt werden kann. Für eine Anwendung des Frameworks an ein konkretes Unternehmen ist, nach einer ausführlichen Analyse der notwendigen Funktionalitäten, dessen Anpassung möglich. Es sei an dieser Stelle betont, daß die Anwendung des Frameworks auf die Arbeitsvorbereitung innerhalb von Gießereien nur ein Anwendungsbeispiel darstellt.

5 Zusammenfassung

Die Einführung einer Entwurfsumgebung zur Unterstützung der Produktentwicklungsprozesse ermöglicht den Unternehmen, die vorhandenen Softwarewerkzeuge für die Produktentwicklung wesentlich effektiver einzusetzen. Dabei werden von diesen Werkzeugen die eigentlichen Anwendungsfunktionalitäten bereitgestellt, die mit Hilfe von Basisfunktionalitäten der Entwurfsumgebung koordiniert werden können. Bekannte Realisierungsstrategien für Entwurfsumgebungen betonen nur einzelne Aspekte, wodurch Defizite innerhalb der Unterstützung auftreten. Im Gegensatz zu diesen Ansätzen werden bei dem in diesem Beitrag vorgestellten Ansatz die auszuführenden Entwurfsprozesse unter Berücksichtigung der erforderlichen Entwurfsdokumente und Werkzeuge in den Mittelpunkt der Betrachtungen gestellt. Für die Umsetzung dieses Ansatzes kommt sowohl für die generelle Entwurfsumgebung, als auch für die Prozeßkomponente ein komponentenbasierter Ansatz zur Anwendung, um auf die sich ständig ändernden Anforderungen von Seiten der Unternehmen an diese Umgebung adäquat reagieren zu können. Grundlage bildet dabei ein Komponentenmodell, mit dem die Spezifizierung von autonomen Komponenten und deren Verbindung zu komplexen Komponenten möglich ist. Durch die Bereitstellung von komponentenübergreifenden Anwendungsmodellen kann die Verwendung von autonomen Komponenten innerhalb eines neuen Gesamtkontextes koordiniert werden. Der Ansatz, welcher dem Frameworkgedanken folgt, wurde mit Hilfe einer Prototypumgebung validiert und theoretisch auf den Bereich der Arbeitsvorbereitung in Gießereien angewendet. Generell muß die Eignung dieses Ansatzes noch in einem realen Unternehmen nachgewiesen werden.

Zukünftige Forschungsarbeiten befassen sich sowohl mit der Weiterentwicklung des Prototyps als auch mit der Ergänzung einzelner Konzepte der Entwurfsumgebung. So ist beispielsweise eine Ergänzung um eine Projektmanagementkomponente erforderlich. Aufgrund der Verwendung eines komponentenbasierten Implementierungsansatzes kann diese zunächst autonom entwickelt und dann „ohne“ Probleme in die Gesamtumgebung eingefügt werden.

Literatur

1. Aalst, W.M.P. van der: Chapter 10: Three Good Reasons for Using a Petri-net-based Workflow Management System. In: Wakayama, T., Kannapan, S., Khoong, C.M., Navathe, S. und Yates, J. (Herausgeber): Information and Process Integration in Enterprises - Rethinking documents, The Kluwer International Series in Engineering and Computer Science, Seiten 161-182. Kluwer Academic Publishers, Norwell, 1998.
2. Abramovici, M.: Informationsmanagement und -logistik mit Engineering-Data-Management. In: Verein Deutscher Ingenieure (Herausgeber): Effiziente Anwendung und Weiterentwicklung von CAD/CAM-Technologien, Band 1289 der Reihe VDI-Berichte, Seiten 303-322, VDI-Verlag, 1996.
3. Allen, P.: Component-based Architecture: A Call for Pragmatism. In: Grundy, J. (Herausgeber): Proceedings of Workshop on Component-based Information Systems Engineering (CBISE98), Working Paper 98/12 (Department of Computer Science, University of Waikato Hamilton, New Zealand), Seiten 5-12, Pisa, Italy, Juni 1998.
4. Allen, R. und Garlan, D.: Formal Connectors. Technischer Bericht CMU- CS-94-115, Carnegie Mellon University, März 1994.
5. Ambos, E., Hofmann, I., Krötzsch, S., Pfisterer, W. und Scheler, R.: Rechnerunterstützte Arbeitsvorbereitung - der Weg in die Zukunft. GIESSEREI 86, 5:53-62, Mai 1999.
6. Baumgarten, B.: Petri-Netze. Wissenschaftsverlag, 1991.
7. Becker-Kornstaedt, U., Hamann, D., Kempkens, R., Rösch, P., Verlage, M., Webby, R. und Zettel, J.: The Spearmint Approach to Process Definition and Process Guidance. In: Maurer, F. (Herausgeber): Workshop on Software Engineering over the Internet (ICSE'98), Kyoto, Japan, April 1998.
8. Booch, G., Rumbaugh, J. und Jacobson, I.: The Unified Modeling Language User Guide. Addison Wesley Longman, Inc., 1999.
9. Burger, C.: Groupware - Kooperationsunterstützung für verteilte Anwendungen. dpunkt.verlag, 1997.
10. Buxton, J.N. und Stenning, V.: Requirements for Ada Programming Support Environments: Stone-man. Technischer Bericht U.S. Department of Defense, Februar 1980.
11. Christie, A.M.: Software Process Automation - The Technology and Its Adoption. Springer-Verlag Inc., 1994.
12. Conradi, R., Fernström, C., Fuggetta, A. und Snowdown, R.: Towards a Reference Framework for Process Concepts. In: Demame, J.C. (Herausgeber): Proceedings of the 2nd European Workshop on Software Process Technology, Band 635 der Reihe Lecture Notes in Computer Science, Seiten 3-17, Trondheim, Norway, September 1992, Springer-Verlag Inc.
13. Eigner, M., Hiller, C., Schindewolf, S. und Schmich, M.: Engineering Database - Strategische Komponente in CIM-Konzepten. Carl Hanser Verlag, München, 1991.
14. Eigner, M. und Schmidt, S.: EDM/PDM und verteilte Kommunikation - Teil I. CAD/CAM-Report - Engineering Magazin für C-Technologien, 18(1):42-47, Januar 1999.
15. Elmagarmid, A.K.: Database Transaction Models for Advanced Applications. Morgan Kaufmann Publishers, San Mateo, CA, 1992.
16. Endig, M.: Ansatz einer Komponenten-basierten Prozeßsteuerung zur Modellierung und Ausführung von Engineeringprozessen. In: Turowski, K. (Herausgeber): Tagungsband zum 1. Workshop „Komponentenorientierte betriebliche Anwendungssysteme“, Seiten 65-71, Magdeburg, März 1999.
17. Endig, M., Jesko, D. und Paul, G.: Konzepte zur Prozeßintegration in Rechnerunterstützten Ingenieur-systemen. PrePrint 17, Otto-von-Guericke-Universität Magdeburg, Dezember 1998.
18. Eversheim, W. und Bochtler, W.: Simultaneous Engineering - Erfahrungen aus der Industrie für die Industrie. Springer-Verlag Inc., Berlin, 1995.
19. Garg, P.K. und Jazayeri, M. (Herausgeber): Process-Centered Software Engineering Environments, Los Alamitos, California, Januar 1995, IEEE Computer Society Press.
20. Gausemeier, J., Hahn, A. und Schneider, W.: Kooperatives Modellieren auf Basis transienter Objekte. In: Ruland, D. [40], Seiten 311-325.
21. Grabowski, H., Anderl, R. und Polly, A.: Integriertes Produktmodell. Beuth Verlag GmbH, Berlin, 1993.
22. Grabowski, H. und Geiger, K.: Neue Wege zur Produktentwicklung. Dr. Josef Raabe Verlags-GmbH, Stuttgart, 1997.
23. Gruhn, V.: Validation and Verification of Software Process Models. Doktorarbeit, Universität Dortmund, 1991.
24. Hasenkamp, U., Kim, S. und Syring, M. (Herausgeber): CSCW - Computer Supported Cooperative Work - Informationssysteme für dezentralisierte Unternehmensstrukturen. Addison Wesley, 1994.

25. Krötzsch, S., Hofmann, I., Ambos, E. und Paul, G.: Using of Process Modelling in the Work Scheduling. In: Yang, Q. (Herausgeber): Proceedings of the ISCA 12th International Conference - Computer Applications in Industry and Engineering, Seiten 127-132, The International Society for Computers and Their Applications, November 1999.
26. Krötzsch, S., Hofmann, I., Kreuzmann, F., Ambos, E. und Paul, G.: Rationelle Informationsbereitstellung bei der Bearbeitung von Angeboten für Gußstücke. GIESSEREI 84, 10:15-21, Mai 1997.
27. Loos, P.: Generic Structures - An Approach To Achieve Adaptable Information Systems. In: Doukidis, G., Galliers, R., Jelassi, T., Krcmar, H. und Land, F. (Herausgeber): Proceedings of the Third European Conference on Information Systems (ECIS'95), Seiten 971-978. Print Epress, Juni 1995.
28. Magee, J., Dulay, N., Eisenbach, S. und Kramer, J.: Specifying Distributed Software Architectures. In: Schäfer, W. und Botella, P. (Herausgeber): 5th European Software Engineering Conference (ESEC'95), Band 989 der Reihe Lecture Notes in Computer Science, Seiten 137-153, Sitges, Spain, September 1995, Springer-Verlag Inc.
29. Maurer, F. und Pews, G.: Flexibles Workflowmanagement für Entwurfsprozesse am Beispiel der Bebauungsplanung. In: Ruland, D. [40], Seiten 256-270.
30. Mertins, K., Stüssenguth, W. und Jochem, R.: Modellierungsmethoden für rechnerintegrierte Produktionsprozesse - Unternehmensmodellierung, Softwareentwurf, Schnittstellendefinition, Simulation. Hanser-Verlag, 1994.
31. Nagl, M. und Westfechtel, B. (Herausgeber): Integration von Entwicklungssystemen in Ingenieurwendungen - Substantielle Verbesserung der Entwicklungsprozesse. Springer-Verlag, 1999.
32. Object Management Group: The Common Object Request Broker: Architecture and Specification Revision 2.0, 1997, OMG Document 97-02-25.
33. Object-Oriented Concepts, Inc.: ORBacus - For C++ and Java, 1999, (Handbuch), www.ooc.com.
34. Owen, J.: STEP - An Introduction. Information Geometers Ltd., 2. Auflage, Juli 1997.
35. Paul, G., Wierschin, H., Paul, R. und Hofmann, M.: Produktdatenverwaltungssysteme - Methode und Werkzeug zur Unternehmensintegration. CIM Management, 11(4):49-54, 1995.
36. Pohl, K.: A Process Centered Requirements Engineering Environment. Doktorarbeit, RWTH Aachen, 1995.
37. Polly, A.: Methodische Entwicklung und Integration von Produktmodellen. Doktorarbeit, Universität Karlsruhe, 1996.
38. Rammig, F.J. und Steinmüller, B.: Frameworks und Entwicklungsumgebungen. Informatik Spektrum, 15(1):33-43, Februar 1992.
39. Redmond, F.E.: DCOM : Microsoft Distributed Component Object Model. IDG Books Worldwide, 1997.
40. Ruland, D. (Herausgeber): Tagungsband CAD'96 - Verteilte und intelligente CAD-Systeme, Kaiserslautern, März 1996, Informatik Xpress 8.
41. Sattler, K.-U.: Tool-Komposition in integrierten Entwurfsumgebungen. Doktorarbeit, Otto-von-Guericke-Universität Magdeburg, 1998.
42. Scheer, A.-W., Nüttgens, M. und Zimmermann, V.: Objektorientierte Ereignisgesteuerte Prozeßketten - Methode und Anwendung. In: Scheer, A.-W. (Herausgeber): Veröffentlichung des Instituts für Wirtschaftsinformatik, Universität des Saarlandes, Band 141, Mai 1997.
43. Schefström, D. und Broek, G. van den: Tool Integration - Environments and Frameworks. Wiley Series in Software Based Systems, John Wiley Ltd, 1992.
44. Seiler, W.: Technische Modellierungs- und Kommunikationsverfahren für das Konzipieren und Gestalten auf der Basis der Modell-Integration. Doktorarbeit, Universität Karlsruhe, 1985.
45. Snodgrass, R.: The Interface Description Language: description and use. Computer Science Press, 1989.
46. Staud, J.: Geschäftsprozeßanalyse mit Ereignisgesteuerten Prozeßketten. Springer-Verlag Inc., 1999.
47. Suhm, A.: Produktmodellierung in wissensbasierten Konstruktionssystemen auf der Basis von Lösungsmustern. Doktorarbeit, Universität Karlsruhe, 1993.
48. Syan, C.S.: Introduction to concurrent Engineering. In: Syan, C.S. und Menon, U. (Herausgeber): Concurrent Engineering - Concepts, implementation and practice, Seiten 3-23, London, 1994, Chapman & Hall.
49. Wasserman, A.I.: Tool Integration in Software Engineering Environments. In: Long, F. (Herausgeber): Software Engineering Environments: Proceedings of the International Workshop on Environments, Band 467 der Reihe Lecture Notes in Computer Science, Seiten 137-149, Berlin, September 1990, Springer- Verlag Inc.