

ANSATZ ZUR NUTZUNG VON TICKET-SYSTEMEN FÜR DIE AD-HOC-GESTALTUNG VON PRODUKTENTWICKLUNGSPROZESSEN

Anna Schneider, Hartmut Krehmer, Christina Stöber

Zusammenfassung

Um die Entwicklungszeiten interdisziplinärer Produkte zu verkürzen und zugleich hochwertige Erzeugnisse zu gewinnen, ist innerhalb der Produktentwicklung eine flexible Prozessunterstützung erforderlich. Die Untersuchungen auf diesem Gebiet zeigen jedoch, dass die Produktentwicklung aufgrund ihrer spezifischen Eigenschaften durch klassische Workflowmanagementsysteme (WFMS) allein nicht unterstützt werden kann [1]. Dies liegt daran, dass die Gestaltung des Prozesses vom System vorweggenommen wird, d.h. dem Entwickler wird ein starrer Prozessverlauf vorgegeben, der unabhängig von der Aufgabe, der Entwicklungssituation, dem zugehörigen Bearbeiter, möglichen Abweichungen und unvorhersehbaren Ereignissen definiert wurde. Trouble Ticket Systeme (TTS) haben sich sowohl in den Bereichen Service und Support als auch als so genannte Bug Tracker (Fehlererfassungs- und Verfolgungssysteme) in der Softwareentwicklung etabliert. Aktivitäten wie Fehlermeldungen, Aufgaben oder Änderungswünsche, allgemein „Tickets“ genannt, werden nach ihrer Erfassung einem Bearbeiter zugewiesen und durchlaufen einen Statusworkflow (z.B. „offen, in Bearbeitung, geschlossen“). Die Tickets werden von der Auslösung bis hin zum Abschluss dokumentiert und geben dank Historisierung Auskunft über ihren Lebenszyklus. Somit wird ein Ticket zu einer elektronischen Datenmappe, aus der jederzeit Informationen wie zum Beispiel der verantwortliche Bearbeiter, der Bearbeitungsstatus, Termine oder Dokumente ersichtlich sind und für Auswertungen zur Verfügung stehen. Auf diese Weise dient ein TTS auch als eine globale Kommunikationsplattform für alle Projektbeteiligten.

Der Ansatz, der in diesem Beitrag vorgestellt wird, ermöglicht eine flexible IT-gestützte Gestaltung des Produktentwicklungsprozesses (PEP). Zielsetzung dabei ist es, einerseits die Prozessüberwachung zu unterstützen, andererseits die Kommunikation über Abteilungen hinweg zu verbessern. Während bei klassischem WFMS-Ansatz die Initiative vom System selbst ausgeht, berücksichtigt dieser Ansatz, dass die Initiative zur individuellen Prozessgestaltung von dem Entwickler selbst übernommen wird. Dadurch wird dem Ingenieur die eigenverantwortliche Gestaltung des detaillierten Prozessverlaufs in Abhängigkeit vom Kontext freigestellt. Anhand eines Beispiels wird erläutert, wie ein speziell für die Gestaltung von Produktentwicklungsprozessen konfiguriertes TTS zur Unterstützung Ad-hoc-Abläufe und zur Verbesserung der Kommunikation im Produktentwicklungsprozess eingesetzt werden kann. Der Beitrag entstand im Rahmen des Forschungsverbundes ForFlow unter Zusammenarbeit der Covum AG und des Lehrstuhls für Konstruktionstechnik der Universität Erlangen-Nürnberg.

1 Workflowmanagementsysteme im PEP

Der PEP verläuft in der Praxis äußerst unstrukturiert, da er sich erst in seinem Verlauf entfaltet. Besonders zu Beginn ist er stark von der individuellen Arbeitsweise und der Kreativität des Bearbeiters geprägt und stellt eine geistig-schöpferische Tätigkeit dar. Laut einer repräsentativen Untersuchung in Industrieunternehmen diverser Branchen sind die Aufnahme, Verarbeitung und Weitergabe von Informationen die wesentlichen Tätigkeiten eines Ingenieurs [2][3]. Nur 11% der Zeit eines Ingenieurs wird für konstruktive Aufgaben aufgewendet,

während sich der Großteil der Zeit auf Verwalten (34%), Kommunizieren (31%) sowie unproduktives Warten auf Entscheidungen, Informationen und Genehmigungen (24%) verteilt. Durch die Festlegung von Lösungen und dem anschließenden Erkenntnisgewinn über neue Randbedingungen ist die Konkretisierung des Produktes abhängig von der Wahl des Lösungsweges. Somit hängen nachfolgende Schritte im PEP stark von den Ergebnissen vorangegangener Schritte ab. Durch das ständige Pendeln zwischen Abstrahieren und Konkretisieren, sowie durch den ständigen Kreislauf aus Bewerten, Entscheiden und Handeln ergeben sich häufig Rücksprünge auf vorangegangene Arbeitsschritte, was sich in einem iterativen Vorgehen niederschlägt. Besonders die zunehmend parallele Konkretisierung von Produkt und zugehörigem Fertigungsprozess sowie die im globalen Wettbewerb verstärkt stattfindende verteilte Entwicklung haben zu einer erheblichen Komplexitätssteigerung des PEP geführt. Gleichzeitig ist damit der Bedarf nach verbesserter Kommunikation, effizienten Benachrichtigungssystemen, zuverlässigen Fortschrittsrückmeldungen sowie nach einer durchgängigen Dokumentation des Prozesses gestiegen.

In der Literatur wird zwischen drei Arten von Workflows unterscheiden, und zwar Production-, Administrative- und Ad-hoc-Workflows [4][5]. Die klassischen WFMS wurden für die Unterstützung von Production- und Administrative Workflows konzipiert. Diese Arten von Workflows werden durch den klar vordefinierten Aufbau sowie eine hohe Wiederholungsfrequenz charakterisiert [6]. Das Ziel war und ist die Standardisierung der Abläufe mit nachfolgender Automatisierung. Feste Arbeitsabläufe können entweder auf einem sehr hohen Abstraktionsniveau oder auf einer sehr detaillierten Ebene erfasst werden [7]. Im ersten Fall gehen sie nicht über die Grobausrichtung des Prozesses auf der Ebene des Projektmanagements hinaus: Die einzelnen Projektphasen werden als Prozessschritte definiert und stellen meist eine Sequenz dar. Für die tägliche Arbeit eines Ingenieurs birgt ein solcher Ablauf keinen hohen Nutzen in sich, weil die Arbeitsabfolge selbst als viel tiefer greifende „Verfeinerung“ eines vorgegebenen Prozessschrittes zu betrachten ist. Die Zertifizierung nach ISO 9000 liefert im PEP-Bereich höchstens Freigabedefinitionen bzw. Angaben zu Meilensteindefinitionen [7]. Wird eine Vorgehensweise sehr detailliert vorgegeben, so bleibt dem Entwickler kaum Raum für die Entfaltung seiner Kreativität. Auch bei der Definition der BPEL (Business Process Execution Language) wurde schlichtweg vergessen, menschliche Aktivitäten in die Notation aufzunehmen. Das Hauptaugenmerk wurde vielmehr auf die so genannte „Orchestrierung“ unterschiedlicher (Web)-Services im Rahmen eines Prozesses gelegt. Mit der in den letzten Monaten neu erschienenen Definition der BEPL4people wird dieses Versäumnis nachgeholt, wobei noch abzuwarten ist, ob sich die Notation als Standard etablieren wird.

Für eine angemessene Unterstützung des PEP, welche die Kreativität des Entwicklers sowie den iterativen Prozessverlauf angemessen berücksichtigt, sind klassische WFMS als nur bedingt geeignet einzustufen. Dies liegt daran, dass hier die Gestaltung des Prozesses systemseitig vorweggenommen wird, d.h. dem Entwickler wird ein fester Prozessverlauf vorgegeben, der unabhängig von der Aufgabe, der Entwicklungssituation, dem zugehörigen Bearbeiter, möglichen Abweichungen sowie von unvorhersehbaren Ereignissen generiert wurde. Selbst die noch relativ seltenen adaptiven WFMS setzen die Definition eines Ausgangsprozesses voraus. Jedoch kann der WFMS-Ansatz auch bei Freigabe- oder Korrekturmechanismen durchaus denkbar oder sogar ratsam sein. Im Rahmen des Engineering Workflows wird vorgeschlagen, solche Freigabeworkflows einzusetzen [7]. Aus oben genannten Gründen muss sich die systemseitige Unterstützung des PEP nicht nur auf die Steuerung und Lenkung des PEP konzentrieren, sondern sich zusätzlich ad-hoc um die Sättigung des Informationsbedarfs sowie um die Unterstützung der Kommunikation kümmern. Da sich die Prozesse in der Produktentwicklung grundlegend von Geschäftsprozessen unterscheiden, die Kreativität der in Teams arbeitenden Ingenieure nicht beeinträchtigt werden darf und sich das Suchen und Vergleichen von Informationen schwierig gestaltet, bedarf es einer flexibleren Vorgehensweise als bei den starren WFMS.

Durch die für die Entwicklung immer komplexerer Produkte notwendigen und im Rahmen der Globalisierung und Internationalisierung zunehmend über Unternehmens- und Ländergrenzen hinweg verteilten Entwicklerteams ist der Abstimmungsbedarf und somit die Notwendigkeit einer reibungslosen Kommunikation und eines regen Informationsaustausches stark gestiegen. Des Weiteren erhalten in diesem Zusammenhang auch die zügige Ermittlung von Ansprechpartnern und Verantwortlichen, die Suche nach bereits vorhandenen Teillösungen sowie die Dokumentation von Wissen einen immer höheren Stellenwert. Eine besondere Bedeutung wird in VDI-Richtlinie 2221 der Modularisierung von arbeitsintensiven Gestaltungsschritten zugeschrieben. Insbesondere bei komplexen Produkten ist diese wichtig, „um eine effiziente Aufteilung der Konstruktionsarbeit zu erleichtern und durch Strukturierung bestimmte Entwicklungsschwerpunkte besser erkennen und lösen zu können“ [8]. Ein modernes WFMS für die Produktentwicklung muss folgende Anforderungen erfüllen:

- Um im Sinne eines kompetenten Auftretens dem Auftraggeber gegenüber die Auskunftsfähigkeit zu verbessern und die Verbindlichkeit der Termineinhaltung zu gewährleisten, muss ein modernes WFMS die Auskunft über Verantwortlichkeiten, das Auffinden kompetenter Wissensträger als Ansprechpartner, die Rollenverteilung, die Kompetenzenorganisation sowie eine bedarfsgerechte Kommunikation und Zugang zu entscheidungsbestimmenden Informationen unterstützen [7].
- Um den Zeit- und Kostenrahmen eines Projektes einhalten zu können, muss dem Entwickler ein umfassender Überblick über die verteilten Tätigkeiten, den Fortschritt sowie die anstehenden Aufgaben im PEP verschafft werden.
- Um die Prozessunterstützung für nachfolgende Entwicklungen zu verbessern, müssen durch die Kontextspeicherung ein verbessertes Wiederauffinden von Informationen und durch die vollständige Dokumentation eine gesteigerte Nachvollziehbarkeit der Entscheidungen ermöglicht werden.

Zur Unterstützung von Ad-hoc-Abläufen werden heute in vielen Projekten E-Mail-Systeme eingesetzt. Die Motivation liegt darin, dass die Bedienung eines E-Mail-Clients einfach und die Verbreitung und Benutzerakzeptanz hoch sind [9]. E-Mail-Systeme wurden jedoch von Grund auf nicht für die Aufgaben einer rechnergestützten Zusammenarbeit geschaffen. Es fehlen Funktionalitäten wie zentrale Datenhaltung, berechtigungsgesteuerte Zugriffsmöglichkeiten, Diskussionsmöglichkeiten, Versionierung der angefügten Daten und Dokumente, Transparenz, Bildung von Relationen, zeitliche Überwachung, Eskalationsszenarien sowie die Verfolgung des Bearbeitungsstatus. Die Behebung all dieser Unzulänglichkeiten erfordert die Integration neuer Erweiterungsmodule; der Prozessgedanke wird nicht unterstützt. Die genannten Tatsachen sprechen eindeutig gegen die Verwendung von E-Mail-Systemen für eine nachvollziehbare Unternehmenskommunikation sowie für die Unterstützung von Workflows. Höchstens bei der Benachrichtigung der Mitarbeiter über neu angetroffenen Aufgaben oder Anfragen bzw. als Kommunikationsmittel mit einem WFMS ist die Verwendung von E-Mail praktikabel. Nichtsdestotrotz werden E-Mail-Systeme und Groupware gerne als Grundlage zur Realisierung von Ad-hoc-Abläufen genutzt.

Die TTS erheben den Anspruch, die Defizite der E-Mail basierten Systeme zu beseitigen. Dabei gilt es, nicht nur aus dem E-Mail-Bereich bekannte Kommunikation in Form von Nachrichten, weitergeleiteten Nachrichten, Nachrichten mit Anhängen etc. in verschiedensten Formaten zu unterstützen, sondern vielmehr eine zentrale Kommunikationsplattform für alle beteiligten Personen zu schaffen, die zur Initiierung, Verwaltung, Dokumentation und Nachvollziehbarkeit der in ihr verwalteten Aktivitäten dient. Außerdem sind Internet-Portale, Diskussionsforen und Webbrowser heute weit verbreitet und einem großen Benutzerkreis zugänglich, was ebenfalls für den Einsatz eines solchen Systems spricht.

2 Einbindung von Trouble-Ticket-Systemen in die Ad-hoc-Gestaltung des Produktentwicklungsprozesses

Im Request for Comments (RFC) 1297 wurde schon 1992 folgende Beschreibung verwendet: „[...] *problems require some kind of problem tracking system, herein referred to as a trouble ticket system. A basic trouble ticket system acts like a hospital chart, coordination the work of multiple people who may need to work on the problem. [...]*“.

Ein Trouble Ticket ist eine Datenmappe, welche zu einem bestimmten Zeitpunkt einer verantwortlichen organisatorischen Einheit oder Rolle zugewiesen wird. Alle anderen beteiligten Personen, die berechtigt sind, mitzuarbeiten, haben Zugriff auf die Informationen dieser Mappe und können sie vervollständigen bzw. sich über den Fortschritt der Bearbeitung informieren. Der Bearbeitungsfortschritt wird durch den Status des Tickets beschrieben. Die Entscheidungen und die Anweisungen für die beteiligten Personen können ebenfalls festgehalten werden. Je nach System kann ein Ticket unterschiedliche Informationen beinhalten. Der Aufbau eines Tickets wird am Beispiel von Covum Request vorgestellt. Ein Ticket kann folgende Attribute beinhalten:

- ID oder Schlüssel zur eindeutigen Identifizierung
- Angaben über den Autor und den Verantwortlichen
- Bearbeitungsstatus und Priorität
- Erstellungsdatum, Änderungsdatum
- evtl. Kategorisierungsinformationen zur besseren Zuordnung und Wiederauffindung
- Berechtigungsinformationen
- Betreff (Titel) des Tickets und Beschreibung des Sachverhaltes

Je nach System kann auch eine Reihe von anderen Informationen vorhanden sein, meistens in Form von weiteren Attributen. Diese Attribute sind frei definierbar und beschreiben je nach Branche und Anwendung des Systems die bestimmten Charakteristiken des Tickets. Folgende Bereiche decken Trouble Ticket Systeme u. a. ab:

- Zentrale Erfassung, Kategorisierung, Verschlagwortung, Speicherung von Tickets
- Dokumentation des Lebenszyklus eines Tickets: Nachvollziehbarkeit und Änderungsverfolgung wie z.B. Statuswechsel, Weiterleitung
- Benachrichtigungsfunktionen
- Dokumentenaustausch, Abbildung von Informationsflüssen und Organisationsstrukturen
- Diskussions- und Kommunikationsmöglichkeiten (z.B. Forum, Chat, Weiterleitung, Benachrichtigung)
- Wiederauffindbare Wissensbausteine
- Eskalation, zeitliche Überwachung, Vertreterkonzepte

Beim Arbeiten in einem TTS müssen sich die zusammenarbeitenden Benutzer an folgende Regel halten: Alles, was einer Erledigung innerhalb eines Projektes bedarf, wird als Ticket verfasst. Alle Entwicklungsobjekte werden also in Form von Tickets in einem TTS angelegt. Da auf diese Weise eine große Menge an Tickets entstehen kann, werden diese nach speziellen Themengebieten bzw. Kategorien sortiert. Der Aufbau solcher Kategorien ist anwendungsspezifisch und kann zum Beispiel an die Produktstruktur angelehnt werden. Nach der Verfassung in einer bestimmten Kategorie wird das Ticket meistens automatisch einem Bearbeiter, dem Kategorienverantwortlichen, zugewiesen. Je nach Projektorganisation kann es aber auch eine bestimmte Person wie ein Projektleiter sein, der zuerst alle Tickets erhält und erstellt, diese priorisiert und dann den am besten dazu qualifizierten Mitarbeitern zuweist.

Die meisten TTS stellen dem administrierenden Anwender die Möglichkeit zur Verfügung, die Bearbeitungszustände in Form von Statusworkflows zu definieren. Ein Ticket kann somit einen Zustandsgraph, bestehend aus zugelassenen Zustandsübergängen, erhalten. Je nach Anwendung können bei jedem Statusübergang die verantwortlichen organisatorischen Einheiten angegeben bzw. die Statusänderungen zugelassen oder verweigert werden. Restriktionen der Form „Der Autor darf sein Dokument nicht selbst freigeben“ können ebenfalls definiert werden. Durch die Angabe von Kommentaren können auch räumlich voneinander entfernte Entwickler die Lösungswege zum Problem diskutieren. Der Lebenszyklus eines Tickets wird von folgenden Aktivitäten beeinflusst:

- Erledigung des geforderten Aufgabenabschnittes durch eine für das Ticket verantwortliche Person
- Änderung des Bearbeitungsstatus mit Benennung des nächsten Bearbeiters
- Arbeitsanweisungen an den nächsten Bearbeiter
- Diskussion der Problemlösung mittels Kommentier- und Zitierfunktionen
- Verbinden mit relevanten Dokumenten, Methoden, ähnlichen Problemtickets
- Terminliche Überwachung, Terminangaben, Eskalationsszenarien

Es ist ersichtlich, dass ein Ticket neben den rein „fachlichen“ Informationen über das Problem und die Problemlösung auch „administrative“ Informationen über die Verantwortlichkeiten, die Erledigungsgeschwindigkeit und die Terminangaben beinhaltet. Durch diese zweite Eigenschaft wird neben einem fachlichen Nachschlagewerk auch für die Erledigung mehr Verbindlichkeit erreicht. Die Ansprechpersonen für das Problem können ermittelt, die Lösungswege können verfolgt und wieder aufgenommen werden. Durch die Verbindung mit Verantwortlichkeiten sehen die Mitarbeiter, dass die Aufnahme der Daten nicht nur „lästiges“ Dokumentieren, sondern vielmehr Bereitstellung und Einholung von relevanten Informationen für sich und andere bedeutet.

3 Beispiel für den Einsatz von TTS

Die Befriedigung des Informationsbedarfs, die Unterstützung der Kommunikation sowie die Nachvollziehbarkeit der Entscheidungen sind wichtige Erfolgskriterien des PEP. Insbesondere bei räumlich verteilt arbeitenden Teams ist der Abstimmungsbedarf und Informationsaustausch besonders hoch. Das Beispiel zeigt anhand eines Auszugs aus der Entwicklung eines Kugelschreibers, wie die Abstimmungen sowie der Datenaustausch zwischen den Beteiligten in einem dokumentierten Ad-hoc-Prozess vollzogen werden können. Auch Defizite bei der Vorgehensweise sowie die Lösungswege werden aufgezeigt.

3.1 Rahmenbedingungen und Basissystem

In VDI-Richtlinie 2221 werden die Rahmenbedingungen einer Produktentwicklung bezüglich der Herkunft der Aufgabenstellung, der Fertigungsart sowie des Neuheitsgrades bestimmt [8]. Folgende Rahmenbedingungen sind für das Beispiel definiert: Bezüglich der Aufgabenstellung handelt es sich um einen Kundenauftrag mit einem festen Pflichtenheft, bezüglich der Fertigungsart handelt es sich um eine Einzelfertigung und bezüglich des Neuheitsgrades handelt es sich um eine Anpassungskonstruktion.

Für das Beispiel wird Covum Request verwendet, da dieses System in seiner Grundauführung für unterschiedliche Einsatzgebiete offen gehalten ist. Durch die Konfiguration des Basissystems erhält der Benutzer eine ganz spezielle Anwendung, die seinen Bedürfnissen angepasst ist. So kann auf Basis eines Kernsystems je nach Bedarf ein Service-System, ein Fehlererfassungs- und Verfolgungssystem oder die Unterstützung von Ad-hoc-Abläufen in einem PEP realisiert werden. Programmiertechnische Erweiterungen, insbesondere Verbindungen zu externen Systemen, sind dabei nicht ausgeschlossen, jedoch können die meisten Features allein durch Konfiguration in einem Administrations-Interface verfügbar gemacht werden. Das System bietet einen rechteckgesteuerten Zugang zu allen Arbeitsobjekten in Form von Tickets. Es erscheint sinnvoll, den Einstieg in das System über die Produktstruktur zu ermöglichen. In dem Beispiel wird für die Organisation der Konstruktionsabläufe folgende vereinfachte Strukturierung verwendet:

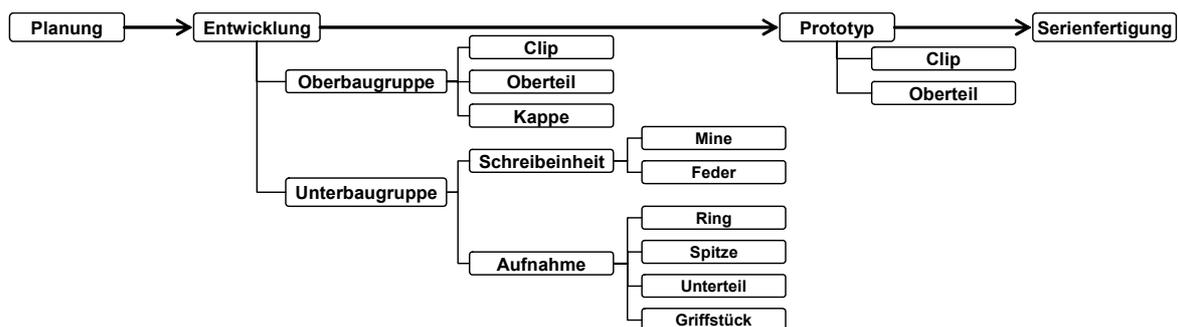


Bild 1: Beispielhafte Konstruktionsablauf-Organisation bei Entwicklung eines Kugelschreibers

Eine zu erledigende Aktivität wird somit bei der Erstellung in eines dieser Themengebiete eingeordnet. In der Festlegung einer geeigneten Detaillierung besteht dabei eine besondere Herausforderung. Allgemein gilt: Je höher in der Themenhierarchie sich das Ticket befindet, umso allgemeiner ist seine Formulierung, umso mehr Konkretisierung muss vorgenommen werden. Ein entsprechend gutes Abstraktions- und Konkretisierungsvermögen wird von den ausführenden Mitarbeitern gefordert.

Während der Entwicklungsphase fallen Informationen über Daten, Entscheidungen und Kommunikationsflüsse innerhalb der Konstruktion an. Als Beispiel wird eine Funktion aus dem Pflichtenheft mit folgender Beschreibung verwendet:

„Eine Unterbaugruppe mit Griff ist zu konstruieren. (Folgende Daten sind dabei zu beachten: Griffstückmerkmale: Rutschfester Griff, Länge der Greiffläche: 8mm, Schwerpunkt: 60mm von Spitze der Mine, Gewicht: 15g.) Bitte benutzen Sie das beigefügte Pflichtenheft für weiterführende Informationen.“

Diese Funktion wird in das entsprechende Themengebiet „Griffstück“ eingetragen und einem Bearbeiter zugewiesen. Dann kann die Bearbeitung mit einem Ad-hoc-Workflow beginnen. Der Bearbeitungsbeginn, der Entwicklungsfortschritt, die Iterationen und der Abschluss einer Aktivität können z.B. durch den Status gekennzeichnet werden. Die Projektbeteiligten haben

miteinander zu vereinbaren, welche Statusübergänge zu verwenden sind. Im Beispiel werden folgende Workflowzustände verwendet:

- Offen: Grobe Darstellung der Anforderungen an die Funktion
- In Bearbeitung: Hierzu gehören die Konkretisierung wie Klären und Präzisieren der Aufgabenstellung
- Test: Integration mit anderen Komponenten, Bewertung
- Schließen die Tests fehl, kann zurück zu „In Bearbeitung“ gesprungen werden
- Geschlossen: Zusammenfassung der Ergebnisse und Abschluss des Tickets

Die Statusübergänge sind folgenderweise definiert:

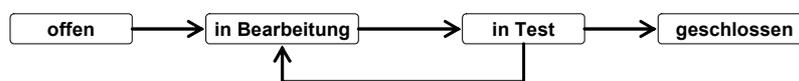


Bild 2: Statusübergänge eines Tickets

Die Arbeit zwischen den einzelnen Statusübergängen gestaltet sich ad-hoc. Das bedeutet, dass dem Bearbeiter ein Freiraum gelassen wird, wie er seine Aufgabe innerhalb eines Status erledigt: Mit welchen Personen er kommuniziert, wie er sich seine Informationen einholt, welche Methoden er benutzt bzw. ob er seine Funktion in mehrere Subfunktionen aufspaltet und diese anderen Mitarbeitern weiterleitet.

Für die Konstruktion der Baugruppe „Kugelschreiber“ werden zwei Objekte benötigt, nämlich die Unterbaugruppe und die Oberbaugruppe, die mit einem Gewinde zusammengeschraubt werden sollen. Durch die Aufspaltung der Aktivität „Unterbaugruppe konstruieren“ in „Schreibereinheit konstruieren“ und „Aufnahme konstruieren“ werden zwei Subaktivitäten erzeugt, welche ihren Bearbeitern zugewiesen werden. Je nach Koordination der Entwicklung können diese Schritte parallel oder sequenziell abgearbeitet werden. Die Entscheidung liegt bei den Entwicklern selbst.

Bei der parallelen Bearbeitung sind mehrere Entwickler beteiligt, die miteinander kommunizieren müssen, damit die beiden Bauteile bei der Montage auch gefügt werden können. Bei der sequenziellen Arbeitsweise mit Beteiligung mehrerer Entwickler kann das Ticket nach der Beendigung des Arbeitsteils des ersten Entwicklers seine Daten zum nächsten Bearbeiter transportieren. Hier kann die Bearbeitung auch mit einem einzigen Ticket erfolgen. Sollten die Daten unvollständig oder unpassend sein, entstehen Iterationen, indem das Ticket wieder zum ersten Bearbeiter weitergeleitet wird. Dank Änderungshistorie werden die Iterationen ersichtlich und bieten eine Grundlage zur Fehlerforschung.

Im Laufe der Entwicklung kommen Dokumente wie Zeichnungen, Dokumentationen, Kommentare sowie weitere Aufgaben hinzu, welche mit dem Ticket verknüpft werden. Es ist wichtig, dass alle für die Erledigung relevanten Informationen verknüpft werden und erreichbar sind. Wenn dies nicht möglich ist, können jedoch anhand der Ticketdaten die Ansprechpartner und verantwortliche Personen ermittelt und kontaktiert werden.

Gängige TTS bieten die Möglichkeit, mehrere Tickets miteinander in meistens frei beschreibbare Relationen der Art „Doppelerfassung“, „Verwandt mit“, „Lösungsbeschreibung von“ etc. zu setzen. Solche Relationen tragen eher Informationen und zeichnen sich aufgrund ihres empirischen Charakters durch fließende Grenzen zwischen den einzelnen Relationsarten aus. Auf diese Weise können Wissensnetze über zusammenhängende relevante

Informationen aufgebaut werden. Aber auch für die Unterstützung des Aufbaus eines Ad-hoc-Workflows erscheint die Verwendung von Referenzierungen sinnvoll.

4 Zusammenfassung und Ausblick

Im vorliegenden Bericht wurde eine Vorgehensweise zur Ad-hoc-Gestaltung eines Workflows unter Einsatz eines TTS im PEP vorgestellt. Im Laufe des PEP kommen viele spontane Abläufe zustande. Als vorteilhaft erweist sich dabei die Verbindung der Ticketerfassung und -verfolgung mit einer Verantwortlichkeit. Somit wird nicht nur ein höherer Verbindlichkeitsgrad sowie bessere Auskunftsfähigkeit bzw. das Auffinden von Ansprechpartnern erreicht. Auch das sonst als „lästig“ empfundene Dokumentieren kann sich einfacher gestalten. Mit Hilfe von Suchfunktionen auf einem zentralen Datenbestand können bereits bearbeitete Themen und Vorgehensweisen gefunden werden. Durch die Referenzierung relevanter Dokumente und Themen können die Zusatzinformationen zügig ermittelt werden. Durch Arbeitslisten hat jeder Mitarbeiter einen besseren Überblick über die anstehenden Aufgaben, auch die Priorisierung der Erledigung ist schnell festlegbar. Der ticketinterne Statusworkflow bietet Auskunft über den Fortschritt der Bearbeitung und kann die Grobschritte vorgeben. Alle genannten Eigenschaften eines TTS bieten eine gute Grundlage für die systemseitige Unterstützung von Ad-hoc-Abläufen im PEP.

Andererseits ist zu beachten, dass der Ad-hoc-Workflow selbst weit über den Statusworkflow eines einzelnen Tickets hinausgehen kann, insbesondere in den Fällen, wo mehrere abstimmsbedürftige Schritte innerhalb eines Tickets von unterschiedlichen Organisationseinheiten erledigt werden müssen. Hierbei sind die Festlegung der Reihenfolge der Schritte sowie die Datenübergabe innerhalb des Ad-hoc-Workflows von hoher Relevanz. Das Ziel ist es, eine Datenstruktur zu erhalten, die es erlaubt, die spontan aufgetretenen Schrittabfolgen nach dem Ablauf zu einem Workflowmodell zu rekonstruieren. Hierbei weisen die TTS noch durch eine Reihe von Defiziten auf.

Dafür sollte neben den heute üblichen rein informationstragenden Referenzen zwischen einzelnen Objekten auch die Berücksichtigung workflowspezifischer Verlinkungen stärkere Ausprägung finden, was detailliertere Datenerfassung für die Visualisierung des Ablaufs von Ad-hoc-Workflows ermöglichen würde. Hierfür fehlen heute Mechanismen und Regeln zur Behandlung von Iterationen und Fallunterscheidungen. Da der Statusworkflow selbst nur ein Zustandsgraph ist und die Entscheidungen entweder gar nicht oder nur in Textform dokumentiert werden, können diese auch nicht nachvollzogen werden. Auch an dieser Stelle sind Optimierungen notwendig.

Der vorwiegend in den klassischen WFMS praktizierte Top-Down-Ansatz (Modellierung, Implementierung, Ausführung) funktioniert aufgrund seiner Starrheit in einem PEP nur bedingt und kann höchstens bei Routinevorgängen wie Dokumentenfreigaben oder Korrekturworkflows eingesetzt werden. Die hier vorgeschlagene Methode bedient sich des Bottom-Up-Ansatzes: Der Workflow entwickelt sich ad-hoc durch Verlinkung einzelner Prozessschritte zum Gesamtablauf. Durch die Historisierung der Arbeitsschritte kann der Verlauf des Prozesses nachvollziehbar gemacht werden. Voraussetzung ist die Erhöhung des Verbindlichkeitsgrades sowie unternehmensinterne Vereinbarungen, die zur Nutzung des Systems getroffen werden müssen.

Die Kombination eines TTS mit einem geeigneten WFMS kann beide Ansätze zusammen bringen. Auf diese Weise erhält den besseren Überblick sowohl auf der Projektmanagementebene als auch auf der Ebene der einzelnen Tätigkeiten. Auch feste Workflows oder Workflowschritte könnten so in beliebigen Arbeitsschritten nach Bedarf herangezogen werden. Die daraus resultierenden Strukturen können z.B. an eine Wissensbasis weitergereicht werden.

Ansatzweise verfügen die TTS bereits über die Möglichkeit, Informationen und Objekte miteinander in Relation zu setzen. Die workflowrelevanten Relationen zwischen den einzelnen Ad-hoc-Schritten erfordern jedoch Differenzierungen, die weit über die einfache Benennung von Referenzarten hinausreichen: Es müssen Mechanismen zur Verfügung gestellt werden, um z.B. die Erledigungsreihenfolge der einzelnen Schritte festzulegen und zu steuern. So kann im Laufe der Bearbeitung interaktiv ein Netzwerk aus Aktivitäten aufgebaut werden, welches nicht nur eine Informationsverknüpfung, sondern auch eine Ausführungsreihenfolge und Bedingungen darstellen kann.

Durch die Zerlegung von Arbeitsschritten in Subaktivitäten können sich die Prozesse ad-hoc gestalten und auch ad-hoc geplant werden. Auf diese Weise können aus Teilen des Ad-hoc-Workflows die Regelmäßigkeiten herausgelöst und vom System in Form von Methoden empfohlen werden. Zuvor sind noch einige Schritte zur Erweiterung eines TTS zu bewerkstelligen. Mindestens folgende Eigenschaften sind dafür erforderlich:

- Aufbau workflowspezifischer Referenzen
- Definition und Visualisierung von Informationsflüssen, Kommunikationsnetzen und Abarbeitungsabfolgen
- Mechanismen zur Behandlung von Entscheidungen, Fallunterscheidungen und Iterationen
- Einbindung von Datentransfer und Backend-Systemen wie PDM und DMS
- Geeignete Darstellung für Benutzer

Im Zusammenhang mit der Nutzung eines solchen Systems ist noch zu erwähnen, dass es in vielen deutschen Unternehmen noch stark an der Bereitschaft mangelt, Wissen zu teilen und zu vermitteln. *„Zu viele Unternehmenskulturen sind nicht durch Offenheit, sondern durch Abschottung geprägt [...] erst müssen die Menschen zusammengebracht werden. Dann erst kommen die Computer dran, denn die technische Vernetzung wirkt nur als Beschleuniger... die Technologien können und müssen die Menschen dabei unterstützen, Knowledge aufzubauen und am besten freiwillig zu verteilen, um dem gesamten Unternehmen und seinen Kunden zu Mehrwert zu verhelfen“* [11].

Jedoch steigt die industrielle Akzeptanz langsam an, vor allem in Bereichen Service und Softwareentwicklung, wobei abzuwarten ist, ob dieser positive Trend auf Bereiche der Produktentwicklung übergreift. Die Aufgabe der Forschung muss es sein, diese Entwicklung durch innovative, praktisch einsetzbare Lösungen voranzutreiben.

5 Literatur

- [1] Paetzold, Kristin: Workflow-Systeme im Produktentwicklungsprozess. In: Meerkamm, H. (Hrsg.): Design for X – Beiträge zum 15. Symposium, 14. und 15. Oktober 2004, Erlangen, Lehrstuhl für Konstruktionstechnik, 2004.
- [2] Uthmann, C.v., Turowski, K.: Workflow-basierte Geschäftsprozessregelung als Konzept für das Management industrieller Produktentwicklungsprozesse; Technical Report, Institut für Wirtschaftsinformatik, Münster 1996.
- [3] Kühberger, Christine: Von der Prozess-Analyse zur Workflow-Unterstützung des Produktentwicklungsprozesses im Maschinenbau, Diplomarbeit, München, 2002.
- [4] Plesums, Charles: Introduction to Workflow, in: Fischer, Layna (Hrsg.): Workflow Handbook 2002, Future Strategies, Lighthouse Point, Florida (USA) 2002.

- [5] Allen, R.: Workflow: An Introduction, in: Fischer, Layna (Hrsg.): Workflow Handbook 2002, Future Strategies, Lighthouse Point, Florida (USA) 2002.
- [6] Goltz, M.: Engineering Workflow – Integriertes Daten- und Prozessmanagement in der unternehmensübergreifenden Produktentwicklung, IMW - Institutsmitteilung Nr. 28 (2003).
- [7] Goltz, M.: Engineering Workflow auf Basis eines objektorientierten Produktmodells, IMW - Institutsmitteilung Nr. 27 (2002); Abgerufen unter http://www.imw.tu-clausthal.de/fileadmin/Bilder/Forschung/Publikationen/Mitt_2002/02_07.pdf.
- [8] VDI 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, Düsseldorf, VDI-Verlag, 1993.
- [9] Huth, Carsten: Groupware-basiertes Ad-hoc-Workflow-Management: Das GroupProcess-System (Konzeption und prototypische Implementierung einer „Collaboration on Demand“-Lösung zur Unterstützung von schwach strukturierten Prozessen in Unternehmen), Paderborn, 2004.
- [10] RFC 1297 - NOC Internal Integrated Trouble Ticket System Functional Specification Wishlist; abgerufen am 23.08.2007, <http://www.faqs.org/rfcs/rfc1297.html>.
- [11] Wettbewerbsvorteile durch vernetzte Entwicklungsstrukturen: Straffe Produktentwicklung durch virtuelle Teams (Pressefach Arbeitskreis Engineering Workflow (ewf), <http://www.koenigonline.de/pressezentrum/ewf/ewf.htm>, abgerufen am 23.08.2007).
- [12] Prengemann, Ulf: Workflow in der industriellen Produktentwicklung am Beispiel von SOFRAME, Vortrag auf dem 4. Forum Vorgangsbearbeitung, Frankfurt, 1996.
- [13] Saatweber, Jutta: Kundenorientierung durch Quality Function Deployment Systematisches Entwickeln von Produkten und Dienstleistungen; ISBN 978-3-936608-77-9 (Abgerufen unter <http://www.symposion.de/qfd/20950101.htm>).
- [14] Schmitt, Reinhard: Unternehmensübergreifender Engineering Workflow: Verteilte Produktentwicklung auf der Grundlage eines parameterbasierten Daten- Und Prozessmanagements: Dissertation, Clausthal-Zellerfeld: Papierflieger, 2001.

Anna Schneider, M.A.
Covum AG
Nägelsbachstraße 49c
91052 Erlangen
Tel.: +49-9131-89-1582
Fax: +49-9131-89-1585
E-Mail: schneider@covum.com
Internet: <http://www.covum.com>

Dipl.-Ing. Hartmut Krehmer
Dipl.-Ing. Christina Stöber
Lehrstuhl für Konstruktionstechnik
FAU Erlangen-Nürnberg
Martensstraße 9, D-91058 Erlangen
Tel: +49-9131-85-23216
Fax: +49-9131-85-23223
E-Mail: krehmer@mfk.uni-erlangen.de
stoerber@mfk.uni-erlangen.de
URL: <http://www.mfk.uni-erlangen.de>