

# Automated Enactment Tracking for Dynamic Workflows

Thomas Sauer  
rjm business solutions GmbH  
68623 Lampertheim  
t.sauer@rjm.de

**Abstract:** Mit Workflow-Systemen können Arbeitsabläufe innerhalb einer Organisation effektiv unterstützt und gesteuert werden. Die beteiligten Personen müssen jedoch meist die ausgeführten Tätigkeiten an ein rechnergestütztes System explizit rückmelden, was in vielen Fällen zu erheblichen Akzeptanz-Problemen führt. Die Vorteile eines Workflow-Systems gehen damit verloren oder verkehren sich gar ins Gegenteil. In der vorgestellten Arbeit wird daher der Ansatz der *automatisierten Ausführungsverfolgung* entwickelt. Durch Auswertung der Daten, die während der täglichen Arbeit anfallen, werden die bearbeiteten Aufgaben auf Basis eines idealisierten Prozessmodells automatisch festgestellt. Der Ansatz ist mit einem Multi-Agenten-System praxistauglich umgesetzt und hat sich in einem eGovernment-Projekt erfolgreich bewiesen.

## 1 Einleitung

Ein Geschäftsprozess ist ein strukturierter Arbeitsablauf innerhalb einer Organisation. Anhand von Automation einzelner Aufgaben oder deren Zusammenspiel kann die Ausführung eines Geschäftsprozesses mit einem *Workflow-Management-System* effektiv unterstützt werden [vdAvH02]. So ist es unter anderem durch die rechnergestützte Weitergabe von Dokumenten möglich, Wartezeiten zu vermeiden, die Qualität von Produkten und Dienstleistungen sicherzustellen oder Mitarbeitern in der Abarbeitung einzelner Aufgaben zu assistieren [zM04]. Um seine Vorteile erbringen zu können, muss das eingesetzte Workflow-Management-System stets darüber Kenntnis haben, welche Aufgaben derzeit ausgeführt werden bzw. bereits abgeschlossen wurden.

Meist wird den Prozessteilnehmern abverlangt, diese Status-Informationen selbst zu erheben, z.B. indem Einträge auf einer elektronischen Aufgabenliste passend abgehakt werden. In vielen Fällen wird jedoch das explizite Erfassen von Rückmeldungen über die durchgeführten Tätigkeiten als ineffizient und störend empfunden, so dass Workflow-Management-Systeme oft nicht optimal genutzt oder sogar gezielt umgangen werden [BBS95]. Damit können Geschäftsprozesse nicht mehr unterstützt werden, und die täglichen Abläufe werden eher behindert als unterstützt.

In der Dissertation *Automated Enactment Tracking for Dynamic Workflows* [Sau10] wird daher der Ansatz zur automatisierten Ausführungsverfolgung von Geschäftsprozessen entwickelt. Durch Auswertung der Daten, die jeder Prozessteilnehmer während seiner täglichen Arbeit erzeugt, werden die bearbeiteten Aufgaben anhand eines idealisierten Prozessmodells automatisch ermittelt. Damit können manuelle Rückmeldungen entfallen, so

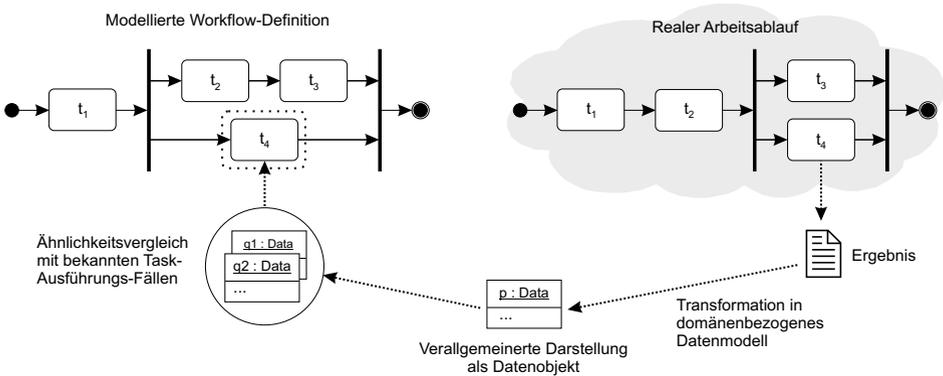


Abbildung 1: Ansatz zur automatisierten Workflow-Ausführungsverfolgung.

dass Workflow-Management-Systeme effektiv eingesetzt werden können und sich die beabsichtigten organisatorischen Vorteile ergeben. Der Ansatz unterstützt insbesondere Vorgänge, die regelmäßig an neue Gegebenheiten wie geänderte Kundenanforderungen angepasst werden müssen. Diese *dynamische Workflows* ermöglichen einen leichten Wechsel von Zuständigkeiten und Prioritäten, der in vielen Domänen unerlässlich ist [BFM<sup>+</sup>06].

Das verwendete Prinzip wird in Abbildung 1 illustriert. Zunächst werden die bei der Abarbeitung einer Aufgabe erzielten Ergebnisse ermittelt. Je nach verwendeten Werkzeugen liegen diese z.B. als Textdokumente oder Datenbankeinträge vor. Die Ergebnisse werden anschließend in eine verallgemeinerte Form überführt, die einen Ähnlichkeitsvergleich mit Angaben erlaubt, die aus vorherigen Bearbeitungen der Aufgabe bereits bekannt sind. Unter der Prämisse, dass eine Aufgabe, die unter ähnlichen Bedingungen erneut abgearbeitet wird, wieder zu ähnlichen Ergebnissen führt, wird bei ausreichender Ähnlichkeit eine neue Bearbeitung abgeleitet. Mit Hilfe eines im Rahmen der vorgestellten Arbeit entwickelten Modells der Workflow-Definition und -Ausführung kann damit schrittweise die Abarbeitung ganzer Geschäftsprozesse ermittelt werden.

Um den Ansatz in der Praxis einzusetzen, wurde das *Progress Information Environment* (PIE) entwickelt. PIE ist als Multi-Agenten-System [Wei00] ausgeführt, um eine möglichst robuste Auswertung von Arbeitsergebnissen zu realisieren. Durch das Zusammenspiel intelligenter, in sich abgeschlossener Software-Agenten können fehlende oder unzureichende Informationen kompensiert werden. Darüber hinaus erlaubt der Systementwurf den flexiblen Umgang mit dynamischen Workflows. PIE wurde erfolgreich im Kontext eines eGovernment-Projekts evaluiert. Das System stellte dabei unter Beweis, dass eine präzise, automatisierte Ablaufverfolgung unter Praxisbedingungen möglich ist und so konkrete Geschäftsprozesse effektiv unterstützt werden können.

Im Abschnitt 2 wird nachstehend zunächst das Modell der Workflow-Definition und -Ausführung beschrieben, das die Basis für die automatische Ausführungsverfolgung bildet. Das PIE-System wird in Abschnitt 3 vorgestellt, und die durchgeführte Evaluation wird in Abschnitt 4 beschrieben. In Abschnitt 5 werden verwandte Arbeiten diskutiert. Eine kurze Zusammenfassung folgt in Abschnitt 6.

## 2 Definition und Ausführung von Workflows

Zur Beschreibung von Geschäftsprozessen stehen einer Organisation eine Vielzahl von Modellierungssprachen zur Verfügung. Insbesondere graphische Sprachen wie Ereignis-gesteuerte Prozessketten (EPK, [KNS92]) oder die Business Process Modeling Notation (BPMN, [Omg06]) sind weit verbreitet. Um zu gewährleisten, dass das Konzept der automatisierten Ausführungsverfolgung unabhängig von der bevorzugten Sprache und der eingesetzten Werkzeuge angewandt werden kann, wurde im Rahmen der hier vorgestellten Arbeit das Modell der abstrakten Workflow-Definition und -Ausführung entwickelt. Konkrete Prozessbeschreibungen können damit in generalisierende *Workflow-Definitionen* eingefasst werden, über die sich die Ergebnisse realer Arbeitsabläufe allgemein beschreiben lassen [SM07]. Eine Workflow-Definition repräsentiert einen Geschäftsprozess als eine Menge von in sich geschlossenen Aufgaben, den *Tasks*, sowie deren mögliche Reihenfolge in Form einer Kontrollflussrelation. Jeder Task stellt eine Aktivität dar, die entweder maschinell ausgeführt werden kann oder von einem menschlichen Prozessteilnehmer übernommen wird. Eine typische Aktivität ist das Zusammenstellen eines Berichtsdokuments, aber auch die Auswahl zwischen zwei verschiedenen Alternativen. Spezialisierte *Trigger-Tasks* können sogar weitere Prozesse anstoßen, ähnlich wie das Aufrufen einer Unterroutine in einer Programmiersprache.

Der Arbeitsablauf innerhalb einer Organisation wird als *Workflow-Ausführung* beschrieben. Dabei werden Tasks wie durch eine Workflow-Definition vorgegeben schrittweise abgearbeitet. Die von den Tasks erzielten Ergebnisse werden als *Datenobjekte* angesehen, wobei das Modell keine Vorgaben hinsichtlich deren Repräsentation macht. In der Anwendung des Modells können Arbeitsergebnisse so in einer für die Anwendungsdomäne passenden Formalisierung betrachtet werden, z.B. als textuelle Beschreibungen, Graph-Strukturen oder Attribut-Werte-Paare. Ist ein Task abgeschlossen, stehen die erzeugten Datenobjekte für die nachfolgenden Tasks bereit, um wiederum weitere Ergebnisse zu erzeugen, usw. Die genaue Abfolge der Task-Ausführungen entlang einer Workflow-Definition bildet eine *Workflow-Instanz*.

Abbildung 2 zeigt ein Beispiel aus der Domäne des geographischen Informations-Managements. Um Karten für eine Ortsbegehung im Stadtgebiet Darmstadt zu erzeugen, wird durch den Trigger-Task „Karten liefern für Darmstadt“ das Ausführen einer passenden

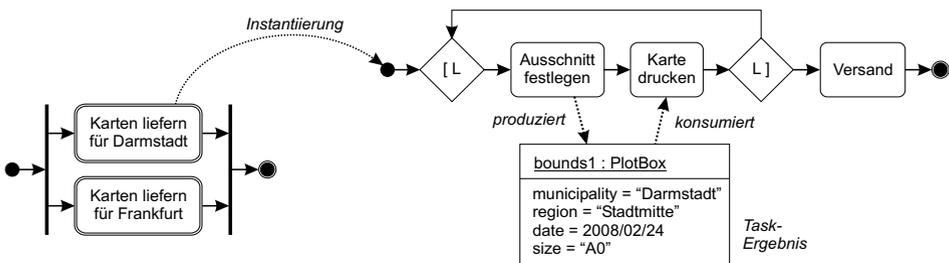


Abbildung 2: Beispiel für eine Workflow-Ausführung.

Workflow-Definition instanziiert. Die dazugehörigen Tasks „Ausschnitt festlegen“ und „Karte drucken“ sind innerhalb eines Schleifen-Konstrukts platziert, gefolgt von Task „Versand“ als letzten Schritt. Damit wird beschrieben, dass zur Bereitstellung von Karten zunächst alle relevanten Bereiche eines größeren Gebiets ausgewählt und gedruckt werden sollen, um anschließend die Ausdrucke gesammelt auf dem Postweg zu verschicken. Als Erstes wird der Task „Ausschnitt festlegen“ aufgeführt. Der dabei identifizierte Bereich wird als Datenobjekt repräsentiert, das auf Basis eines domänenbezogenen Datenmodells den Kartenausschnitt detailliert beschreibt. Das Datenobjekt wird anschließend von dem nachfolgenden Task „Karte drucken“ bei dessen Ausführung verwendet, um entsprechend des Ausschnitts eine Karte zu drucken.

Der aktuelle Fortgang einer Workflow-Abarbeitung wird über einzelne Task-Zustände beschrieben, die sich aus den vorliegenden Daten ergeben. Solange keine Datenobjekte als Task-Ergebnisse vorliegen, wird ein Task als „inaktiv“ angesehen. Sobald ein Task ein Zwischenergebnis produziert hat, aber der Kontrollfluss noch nicht an einen Nachfolger-Task weitergereicht wurde, gilt der Task als „aktiv“. Liegen die Endergebnisse vor, und die Kontrolle wurde an Nachfolger-Tasks übergeben, wird der Task als „abgeschlossen“ angesehen. Da es möglich ist, dass eine Workflow-Definition Schleifen enthält, kann ein Task die einzelnen Zustände mehrfach durchlaufen. Zur eindeutigen Beschreibung einer Bearbeitungssituation wird daher neben den aktuell vorliegenden Datenobjekten und den Task-Zuständen für jeden Task auch eine Wiederholungszahl angegeben.

### 3 Das Progress Information Environment

Das Konzept der automatischen Ablaufverfolgung wird mit dem *Progress Information Environment* (PIE) umgesetzt [SMW08]. Um praxistauglich zu sein, muss das System flexibel mit dynamischen Workflows umgehen können, die z.B. aufgrund von geänderten Kundenwünschen kurzfristig angepasst werden. Darüber hinaus muss PIE in der Lage sein, eventuell fehlende oder ungenaue Daten zu kompensieren. So liegen eventuell für Aufgaben, die aktuell in Bearbeitung sind, zunächst nur Zwischenergebnisse vor, die später vervollständigt werden. Aufgrund dieser Anforderungen ist PIE als Multi-Agenten-System (MAS, [Wei00]) ausgeführt. Abbildung 3 zeigt den grundsätzlichen Systemaufbau. Es werden insgesamt vier verschiedene Arten von Software-Agenten eingesetzt, die in entsprechenden Schichten gegliedert sind. Eine fünfte, unterste Schicht wird durch die Informationssysteme gebildet, wie sie von den Mitarbeitern der jeweiligen Organisation im Arbeitsalltag eingesetzt werden. Derartige Informationssysteme umfassen z.B. Datenbanken oder Systeme zur Dokumentverwaltung. Auf der darüber liegenden Schicht ermitteln *Sensor-Agenten* die mit den Informationssystemen erstellten oder veränderten Daten und transformieren diese in formalisierte Datenobjekte. Auf der nächsten Schicht werten *Task-Agenten* aus, ob die erkannten Datenobjekte auf die Ausführung einzelner Tasks hinweisen. Darüber prüfen *Instanz-Agenten*, zu welchen Workflow-Instanzen diese Ausführung zugeordnet werden kann. Schließlich erzeugen *Workflow-Agenten* eine konsistente Sicht auf den aktuell vorliegenden Abarbeitungszustand hinsichtlich der vorliegenden Workflow-Definitionen.

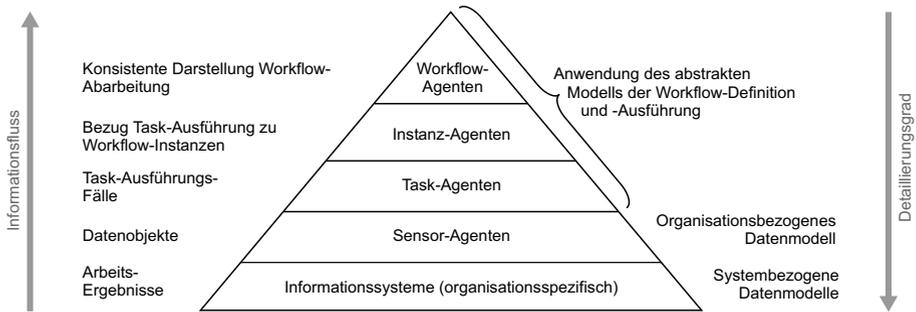


Abbildung 3: Die PIE-Systemarchitektur.

### 3.1 Sensor-Agenten

Sensor-Agenten bilden die Daten eines spezifischen Informationssystems auf Konzepte ab, die auf höheren PIE-Schichten von Task-, Instanz- und Workflow-Agenten verstanden werden können. Wenn ein Mitarbeiter z.B. die in Abbildung 2 gezeigte Aufgabe „Ausschnitt festlegen“ ausführt, wird der ermittelte Kartenausschnitt in einer Datenbank abgespeichert. Ein entsprechender Sensor-Agent transformiert die dazugehörigen Datensätze in ein Datenobjekt von Typ „PlotBox“, um die Arbeitsergebnisse in verallgemeinerter Form auszudrücken. Die für PIE interessanten Systeme umfassen alle rechnergestützten Dienste, Verzeichnisse und Ablagen, die im Arbeitsalltag verwendet werden. Eine redundante Abdeckung einzelner Systeme ist möglich und erwünscht.

### 3.2 Task-Agenten

Für jeden der in den Workflow-Definitionen einer Organisation aufgeführten Tasks ist im PIE-System ein eigenständiger Task-Agent vorgesehen. Task-Agenten nutzen das Prinzip des *Fallbasierten Schließens* (Case-Based Reasoning, [BAM<sup>+</sup>09]), um die von Sensor-Agenten ermittelten Datenobjekten auszuwerten. Fallbasiertes Schließen ist ein etabliertes Problemlösungsverfahren, das zur Lösung eines neuen, aktuell vorliegenden Problems die bekannte Lösung eines ähnlichen, bereits früher gelösten Problems heranzieht. Um das Problem zu lösen, eine Task-Ausführung ausschließlich anhand der vorliegenden Datenobjekte festzustellen, werden diese daher mit Datenobjekten verglichen, die in schon bekannten Bearbeitungssituationen enthalten sind. Jede dieser Situationen repräsentiert einen *Task-Ausführungs-Fall*, der eine Menge von Datenobjekten mit einem konkreten Task-Zustand und einer konkreten Wiederholungszahl korreliert. Die Gesamtheit aller bekannten Task-Ausführungs-Fälle bildet die *Fallbasis* eines Task-Agenten und stellt die Erfahrung dar, die hinsichtlich der Ausführung eines Tasks bereits gesammelt wurde.

Wird ein Task unter ähnlichen Bedingungen wie zuvor ausgeführt, werden erneut ähnliche Ergebnisse erwartet. Sind die Datenobjekte, die ein Task-Agent von Sensor-Agenten

erhalten hat, ähnlich zu denen, die in einem Fall aus der Fallbasis enthalten sind, nimmt der Task-Agent entsprechend an, dass ein Task wie vom Fall beschrieben erneut ausgeführt wurde. Wurde z.B. das "PlotBox"-Datenobjekt wie in Abbildung 2 gezeigt zuvor im Zusammenhang mit dem Ausführungszustand „abgeschlossen“ des Tasks „Ausschnitt festlegen“ beobachtet, und ein ähnliches Datenobjekt wird gemeldet, zieht der zuständige Task-Agent den Schluss, dass der Task erneut ausgeführt wurde und wieder den Zustand „abgeschlossen“ erreicht hat. Zur Beschreibung der neuen Situation wird aus dem bekannten, in der Fallbasis aufgeführten Task-Ausführungs-Fall und den von den Sensor-Agenten gemeldeten Datenobjekten eine neue Kombination von Datenobjekten, Task-Zustand und Wiederholungszahl gebildet. Ein Task-Agent gibt diesen neuen Task-Ausführungs-Fall anschließend an die Agenten auf den höheren PIE-Schichten weiter, welche die Angaben auf Plausibilität überprüfen und gegebenenfalls korrigieren. Sofern der neue Fall tatsächlich zur Beschreibung einer Workflow-Ausführung beiträgt, wird dieser schließlich in die Fallbasis des Task-Agenten aufgenommen.

### 3.3 Instanz-Agenten

Eine Workflow-Definition kann mehrfach gleichzeitig ausgeführt werden, z.B. indem Kollegen parallel zueinander verschiedene Kundenanfragen bearbeiten. Der jeweilige Ausführungspfad wird als Workflow-Instanz beschrieben. Das PIE-System nutzt für jede Workflow-Definition einen Instanzen-Agenten, um festzustellen, welche der Ausführungspfade durch eine von Task-Agenten festgestellte Task-Ausführung potentiell fortgesetzt werden. Dazu wird eine *Distanz* zwischen jedem Task-Ausführungs-Fall, der von einem Task-Agenten übermittelt wurde, und jeder der aktuell bekannten Workflow-Instanzen berechnet. Je kürzer die Distanz, desto eher kann eine Instanz einem Fall zugerechnet werden. Die Distanz-Berechnung erfolgt mittels verschiedener *Distanz-Maße*.

**Anzahl benötigter Schritte.** Ein erstes Distanzmaß ermittelt die Anzahl von Zustands-Transitionen, die benötigt wird, um ausgehend von der zuletzt für eine Workflow-Instanz bekannten Ausführungssituation zu der im Fall beschriebenen Task-Ausführung zu gelangen. Wenn z.B. im Rahmen einer Workflow-Ausführung entsprechend der in Abbildung 2 gezeigten Workflow-Definition zuletzt der Task „Ausschnitt festlegen“ als „aktiv“ beschrieben wurde, und ein Task-Agent hat festgestellt, dass der Task „Karte drucken“ nun Zustand „abgeschlossen“ im Rahmen einer zweiten Schleifeniteration erreicht hat, ermittelt das erste Distanzmaß die Zahl der Zustands-Transitionen, die benötigt werden um (1) den Task „Ausschnitt festlegen“ abzuschließen, (2) Task „Karte drucken“ zum ersten Mal auszuführen, (3) Task „Ausschnitt festlegen“ zu wiederholen und um schließlich (4) Task „Karte drucken“ zum zweiten Mal auszuführen.

**Bezugnahme auf Instanziierungs-Kriterien.** Üblicherweise werden die Kriterien, die zum Bilden einer Workflow-Instanz geführt haben, durch einen Trigger-Task festgelegt. In Abbildung 2 gibt z.B. der Trigger-Task „Karten liefern für Darmstadt“ vor, dass die Workflow-Ausführung mit Bezug auf das Stadtgebiet Darmstadt geschehen soll. Entsprechend werden Datenobjekte, die dieses Gebiet referenzieren, durch ein Distanzmaß mit der entsprechenden Instanz in Verbindung gebracht. Die Distanz wird als umso geringer

festgelegt, je mehr sich die Datenobjekte im Task-Ausführungs-Fall und die Instanziierungskriterien ähneln.

**Betrachtung des Workflow-Kontexts.** Ein weiteres Distanzmaß vergleicht die Datenobjekte eines Task-Ausführungs-Falls mit denen, die bereits im Zusammenhang mit der Workflow-Instanz stehen. Wenn ein Task-Ausführungs-Fall Datenobjekte umfasst, die den bereits erarbeiteten Workflow-Ergebnissen gleichen, wird angenommen, dass sich die im Fall ausgedrückte Task-Ausführung auf ähnliche Aspekte wie die bisherige Workflow-Ausführung bezieht. Eine hohe Ähnlichkeit wird als geringe Distanz umgesetzt.

Durch Kombination der einzelnen Maße wird eine Gesamt-Distanz ermittelt. Der Instanz-Agent nimmt für all diejenigen Paare von Task-Ausführungs-Fällen und Workflow-Instanzen einen Bezug zueinander an, deren Gesamt-Distanz minimal ist. Diese Paare minimaler Distanz werden zur weiteren Prüfung an die nächste PIE-Systemschicht übermittelt. Wenn die Gesamt-Distanz ein zuvor festgelegtes Maximum überschreitet, beschreibt der betreffende Task-Ausführungs-Fall eventuell den Beginn einer neuen Workflow-Ausführung. Der Instanz-Agent schlägt dann eine neue, von bisherigen Workflow-Ausführungen unabhängige Workflow-Instanz vor. Zusammen mit dem Task-Ausführungs-Fall wird dieser Vorschlag ebenfalls zur nächsten PIE-Schicht gemeldet.

### 3.4 Workflow-Agenten

Auf der obersten PIE-Schicht überprüfen Workflow-Agenten, ob die von anderen Systemteilen erhobenen Angaben als gültige Workflow-Ausführung interpretiert werden können. Jeder Workflow-Agent ist für eine bestimmte Workflow-Definition zuständig, und begutachtet alle Paare von Task-Ausführungs-Fällen und Workflow-Instanzen, die von Instanz-Agenten bereitgestellt wurden. Für jeden der Fälle werden die Workflow-Ausführungsschritte ermittelt, um durch Fortsetzung der dazugehörigen Instanz die im Fall beschriebene Situation zu erreichen. Dabei werden auch eventuell fehlende Schritte ergänzt. Wenn z.B. in einer Instanz der in Abbildung 2 dargestellten Workflow-Definition zuletzt der Task „Ausschnitt festlegen“ ausgeführt wurde und der Task-Ausführungs-Fall die Abarbeitung von Task „Versand“ beschreibt, simuliert der Workflow-Agent einen Ausführungsschritt für den dazwischen liegenden Task „Karte drucken“. Damit ist sichergestellt, dass die erkannte Workflow-Abarbeitung zu den Workflow-Definitionen konform bleibt. Die Workflow-Ausführung mit der geringsten Anzahl an notwendigen Simulations-Schritten wird schließlich zur Anzeige des aktuellen Ausführungszustands genutzt.

## 4 Evaluation

Das PIE-System wurde mit dem am Lehrstuhl für Wirtschaftsinformatik II der Universität Trier entwickelten Framework *Collaborative Agent-based Knowledge Engine* (CAKE, [BFM<sup>+</sup>06]) prototypisch implementiert. PIE wurde im Rahmen des eGovernment-Projekts *DenkXweb* erfolgreich evaluiert, das bei der rjm business solutions GmbH im

Auftrag einer hessischen Landesbehörde durchgeführt wird. Das Langzeit-Projekt hat zum Ziel, alle Kulturdenkmäler in Hessen in einem Online-Geoinformationssystem verbindlich nachzuweisen. Damit dies gelingt, müssen zehntausende Gebäude, Straßenzüge und Ortslagen exakt auf Basis von amtlichen Liegenschaftskarten ausgezeichnet werden. Zur effizienten Projektbearbeitung werden eine Reihe von bewährten Verfahren eingesetzt, die als Workflow-Definitionen modelliert wurden. Die Instanziierung der Workflow-Definitionen erfolgt dynamisch entsprechend der Kundenanforderungen.

Das PIE-System wurde in praxisnahen Fallstudien konkret erprobt. Zur Abdeckung der bei ihm verwendeten Software-Infrastruktur wurden 8 verschiedene Sensor-Agenten erstellt, und entsprechend der Workflow-Definitionen wurden 35 Task-Agenten sowie 12 Instanz- und Workflow-Agenten konfiguriert. In einer Fallstudie im Umfang von über 140 Workflow-Ausführungsschritten wurden 75% aller Situationen korrekt erkannt, d.h. die Workflow-Ausführungsverfolgung ist mit einem hohen Maß an Präzision erfolgt. Bezogen auf die Task-Ausführungs-Fälle, die in den Fallbasen der Task-Agenten vorlagen, wurde sogar nahezu das Optimum erreicht (95%). Das heißt, fast alle Workflow-Ausführungsschritte, die einer bereits früher durchgeführten Task-Ausführung ähneln, wurden korrekt erkannt. In einer weiteren Fallstudie stellte das PIE-System unter Beweis, dass auch nach Änderungen in Workflow-Definitionen eine präzise Erkennung der Bearbeitungssituation möglich ist.

## 5 Verwandte Arbeiten

Die Beobachtung und Analyse von Arbeitsabläufen wird in zahlreichen Forschungsgebieten und für viele Anwendungsdomänen diskutiert. So wird z.B. im Bereich des Software-Engineering die automatische Auswertung von Bearbeitungsergebnissen als *Software Project Telemetry* betrachtet [JKP<sup>+</sup>05]. Anhand der Anwendung von Software-Metriken wird der Stand von Entwicklungs-Prozessen bewertet, so dass Fehlentwicklungen frühzeitig erkannt und behoben werden können. Ähnlich dem PIE-Systementwurf wird der Einsatz von Software-Sensoren vorgeschlagen, um die Verwendung von Entwicklungswerkzeugen zu beobachten und für eine weitere Auswertung zusammenzutragen. Eine nähere Identifikation der ausgeführten Arbeitsschritte in einer Prozessbeschreibung erfolgt jedoch nicht.

In [Sch06] wird ein System zur Einschätzung der aktuellen Arbeitssituation von Mitarbeitern in einem Unternehmen vorgestellt, um diesen in der Abarbeitung von Aufgaben durch eine pro-aktive Informationsbereitstellung zu assistieren. Es wird beobachtet, wie die am Arbeitsplatz eingesetzten Software verwendet wird. Aus erkannten Einzelaktivitäten wie dem Öffnen von Dokumenten oder dem Abrufen von E-Mails wird anschließend ein Arbeitskontext abgeleitet und die dazu passende Information ermittelt. Die Mitarbeiter werden damit individuell für einzelne Tätigkeiten unterstützt. Das Ermitteln eines organisationsweiten Bearbeitungsstands der laufenden Geschäftsprozesse, wie es bei der mit PIE umgesetzten automatisierten Ausführungsverfolgung geschieht, ist nicht vorgesehen.

Das in [CB07] erläuterte Konzept verfolgt die Ausführung von Prozessen durch die Auswertung großer Datenströme, wie sie von Systemüberwachungs- und Firewall-Systemen

erzeugt werden. Basierend auf einem automatentheoretischen Prozessmodell wird automatisch ermittelt, ob neue eintreffende Daten einen Zustandsübergang erlauben und so die weitere Ausführung eines Prozesses erklären. Das Konzept sieht jedoch im Gegensatz zur automatisierten Ausführungsverfolgung von Geschäftsprozessen keine Unterstützung dynamischer Arbeitsabläufe vor.

Die automatisierte Ausführungsverfolgung ermöglicht eine weitreichende Überprüfung und Optimierung von Geschäftsprozessen, z.B. durch die Anwendung von *Workflow Mining* [vdAWM04]. Durch eingehende Analyse der tatsächlichen Abfolge der Aufgabenbearbeitung wie mit dem PIE-System ermittelbar werden dabei Workflow-Definitionen aus der täglichen Praxis abgeleitet. Es kann so sichergestellt werden, dass Prozessbeschreibungen und Ausführungsrealität weiterhin übereinstimmen.

## 6 Zusammenfassung

In der Dissertation *Automated Enactment Tracking for Dynamic Workflows* [Sau10] wird der neuartige Ansatz der automatisierten Ausführungsverfolgung von Geschäftsprozessen vorgestellt. Basierend auf einem Modell der abstrakten Workflow-Definition und -Ausführung wertet der Ansatz die Ergebnisse aus, die bei der alltäglichen Bearbeitung von Aufgaben anfallen. Die aktuelle Bearbeitungssituation von Geschäftsprozessen kann so ohne explizite Statusmeldungen der Mitarbeiter ermittelt werden. Die sonst oft beobachteten Akzeptanz-Probleme werden vermieden, so dass Workflow-Management-Systeme effektiv eingesetzt werden können.

Die automatisierte Ausführungsverfolgung wird mit dem Progress Information Environment (PIE) praxistauglich umgesetzt. PIE ist als Multi-Agenten-System ausgeführt, so dass robust auf fehlende oder unzureichende Information reagiert wird. Der Systementwurf ermöglicht zudem den flexiblen Umgang mit dynamischen Arbeitsabläufen, die regelmäßig an neue Anforderungen angepasst werden müssen. PIE wurde im Rahmen eines komplexen eGovernment-Projekts erfolgreich evaluiert.

## Literatur

- [BAM<sup>+</sup>09] Ralph Bergmann, Klaus-Dieter Althoff, Mirjam Minor, Meike Reichle und Kerstin Bach. Case-Based Reasoning: Introduction and Recent Developments. *Künstliche Intelligenz*, 23(1):5–11, 2009.
- [BBS95] John Bowers, Graham Button und Wes Sharrock. Workflow from Within and Without: Technology and Cooperative Work on the Print Industry Shopfloor. In *Proceedings of the 4th European Conference on Computer-Supported Cooperative Work (ECSCW'95)*, Seiten 51–66. Kluwer, 1995.
- [BFM<sup>+</sup>06] Ralph Bergmann, Andrea Freßmann, Kerstin Maximini, Rainer Maximini und Thomas Sauer. Case-Based Support for Collaborative Business. In *Proceedings of the 8th European Conference on Case-Based Reasoning (ECCBR'06)*, Jgg. 4106 of LNCS, Seiten 519–533. Springer, 2006.

- [CB07] George Cybenko und Vincent H. Berk. Process Query Systems. *Computer*, 40(1):62–70, 2007.
- [JKP<sup>+</sup>05] Philip M. Johnson, Hongbing Kou, Michael Paulding, Qin Zhang, Aaron Kagawa und Takuya Yamashita. Improving Software Development Management through Software Project Telemetry. *IEEE Software*, 22(4):76–85, 2005.
- [KNS92] G. Keller, M. Nüttgens und A.W. Scheer. Semantische Prozessmodellierung auf der Grundlage Ereignisgesteuerter Prozessketten (EPK). *Veröffentlichungen des Instituts für Wirtschaftsinformatik*, Heft 89, 1992.
- [Omg06] Business Process Modeling Notation (BPMN) Specification. Bericht, Object Management Group, 2006. Document Number dtc/06-02-01.
- [Sau10] Thomas Sauer. *Automated Enactment Tracking for Dynamic Workflows*. Dissertation, Universität Trier, 2010.
- [Sch06] Sven Schwarz. A Context Model for Personal Knowledge Management. In *Modeling and Retrieval of Context (MRC 2005) Revised Selected Papers*, Jgg. 3946 of LNCS, Seiten 18–33, 2006.
- [SM07] Thomas Sauer und Kerstin Maximini. Supporting Workflow Management by Automated Enactment Tracking. In *Proceedings of the 4th Conference on Professional Knowledge Management*, Seiten 109–118, 2007.
- [SMW08] Thomas Sauer, Mirjam Minor und Sascha Werno. An Agent-oriented System for Workflow Enactment Tracking. In *Proceedings of the 17th Workshops on Enabling Technologies: Infrastructures for collaborative enterprises (WETICE'08)*, Seiten 235–240. IEEE Computer Society, 2008.
- [vdAvH02] Wil M. P. van der Aalst und Kees van Hee. *Workflow Management: Models, Methods and Systems*. MIT Press, 2002.
- [vdAWM04] Wil M. P. van der Aalst, A.J.M.M. Weijters und Laura Maruster. Workflow Mining: Discovering Process Models from Event Logs. *Transactions on Knowledge and Data Engineering*, 16(9):1128–1142, 2004.
- [Wei00] Gerhard Weiss, Hrsg. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. MIT Press, 2000.
- [zM04] Michael zur Muehlen. Organizational Management in Workflow Applications - Issues and Perspectives. *Information Technology and Management*, 5(3):271–291, 2004.



**Thomas Sauer** wurde am 18. April 1977 in Bad Dürkheim geboren. Von 1997 bis 2002 studierte er Informatik mit Nebenfach Mathematik an der Universität Kaiserslautern. Nach seinem Abschluss als Diplom-Informatiker trat er 2003 in die rjm business solutions GmbH als Mitglied der Geschäftsführung und als Mitgesellschafter ein. Er verantwortet seitdem die Bereiche Software-Entwicklung und IT-Dienstleistungen. Im Rahmen einer in 2004 begonnenen Forschungskooperation mit dem Lehrstuhl für Wirtschaftsinformatik II an der Universität Trier promovierte er in 2010 parallel zu seiner beruflichen Tätigkeit zum Dr. rer. nat. Seine Forschungsinteressen umfassen neben allen

Aspekten des Geschäftsprozessmanagements insbesondere die Unterstützung von Unternehmen durch agile Workflows.