

Business Process Management-Systeme

Wo Geschäftsprozesse und IT zusammenwachsen



Bild: Pixelio.de

Workflow- und Business Process Management-Systeme sollen komplexe Geschäftsprozesse möglichst vollständig und durchgängig abbilden. Eine gewaltige Aufgabe, deren Für und Wider abzuwägen ist. Welche Methoden, Standards und Tools für das Business Process Management zur Verfügung stehen, stellt dieser Beitrag vor.

> Was verbirgt sich hinter dem Begriff „Business Process Management-Systeme“, kurz BPMS? Generell lassen sich BPMS folgendermaßen beschreiben: Ein BPMS ist ein allgemeines IT-System, das in der Lage ist, auf Basis eines ausdrücklichen Prozessmodells operative (Geschäfts-) Prozesse umzusetzen, ablaufen zu lassen und diese zu managen.

Ein BPMS bietet also die Möglichkeit, Geschäftsprozesse IT-basiert abzubilden, um mittels eines computergestützten Workflows eine Prozessoptimierung zu erreichen. Was unterscheidet damit ein BPMS von einem sogenannten Workflow-System beziehungsweise einem Workflow-Management-System? Der Übergang ist fließend – der Begriff Workflow-Management wird jedoch mit einem technischen Hintergrund verknüpft und ist mehr als (technische) Basis für ein BPMS zu sehen.

Für Prozessverantwortliche, die sich mit dem Gedanken an die Einführung eines BPMS tragen, sollte klar sein, dass vor der Frage „Welches BPMS ist das Geeignets-te?“ die Frage kommt „Habe ich überhaupt

Vorteile durch den Einsatz eines BPMS?“. BPMS wollen zwei oftmals sehr getrennte Weltanschauungen zusammenführen, nämlich die IT-Sicht und die Prozesssicht auf ein Unternehmen oder einen seiner Teilbereiche. Diese Sichtweisen sind oft nicht einfach zu vereinen, da die Prozessstrukturen meist nicht optimal für IT-Umsetzungen geeignet sind, genauso wie sich einfach zu implementierende Lösungen häufig nur wenig für einen Gesamtprozess eignen. Aus letzterem ergibt sich für viele Unternehmen das Problem, dass über einen Prozess hinweg viele kleine Einzel- oder Inselanwendungen eingesetzt werden müssen, die in der Praxis etwa Synchronisierungsprobleme nach sich ziehen.

Der Ansatz für die Frage, ob sich ein BPMS überhaupt lohnt, geht also ein Stück weit von der Größe und Komplexität des abzubildenden Prozesses aus. Hindert der durch die Einführung eines BPMS entstehende Bürokratieaufwand die Mitarbeiter an ihrer Arbeit, sinkt die Produktivität, ganz zu schweigen von dem Problem der allgemeinen Akzeptanz.

Geht es in dem Zielprozess hingegen um die Ersetzung vieler Einzelschritte und eventuell auch mehrerer kleinerer Programme zur Vereinfachung von Datenverwaltung, so bietet sich ein BPMS durchaus an.

BPMS bieten die Möglichkeit, die beiden Sichten näher zusammenzubringen, indem der Implementierungsaufwand gesenkt wird. Grund dafür ist, dass die meisten BPMS bereits Grundimplementierungen für die Basisfunktionen und Steuerung der Workflows besitzen, genauso wie die damit zusammenhängende Benachrichtigungs- und Berechtigungslogik. Auf der anderen Seite werden die Planer der Prozesse stärker in Richtung der IT gerückt, da sie die Möglichkeit haben, den Prozess mit einem grafischen Tool selbst zu planen und später anzupassen. Womit sie auch schon Teil der eigentlichen Umsetzung sind. Das soll allerdings nicht heißen, dass ein BPMS in Standardinstallation das Ende aller Probleme ist, es bietet lediglich eine Grundlage und einen Ausgangspunkt für Eigenumsetzungen.

Welches BPMS ist das Richtige?

Ist die Frage nach der Einführung eines BPMS geklärt, steht der Interessierte zwangsläufig vor einer erstmal unübersichtlichen Anzahl von Systemen und Prozessdefinitionssprachen. Die untenstehende Übersicht (Bild 1) vermittelt einen ersten Überblick über die Definitionssprachen.

Angesichts der Komplexität des Themenfelds stellt sich die Frage, „welche dieser Definitionen am geeignetsten ist“. Ähnlich wie auch bei den BPMS ist diese Frage nicht eindeutig zu beantworten, jede Definition hat ihre Vor- und Nachteile. Exemplarisch zeigt Bild 2 einen Vergleich zwischen den zwei weit verbreiteten Sprachdefinitionen XPD L und BPEL.

Verfügbare Anwendungen und die Open Source-Frage

Für BPMS gibt es sowohl Open als auch Closed Source-Systeme, genauso wie verschiedene Arten von Frameworks für Eigenentwicklungen. Bild 3 zeigt einige aktuell verfügbare BPMS sowie einige Workflow Management Systeme/Engines, die zur Interpretation von Prozesssprachen

eingesetzt werden können und rudimentäre Funktionen bieten.

Eine Aussage darüber, welches Business Process- oder Workflow Management-System das Richtige ist, kann hier nicht getroffen werden. Jedes der genannten Systeme bietet Vor- und Nachteile, die jeweils individuell geprüft werden müssen.

Finanziell betrachtet, liegen JBoss jBPM und das Intalio BPMS als Open Source-Lösungen klar vor den anderen Systemen, bei denen mehrere Zehntausend Euro Lizenzgebühren pro CPU anfallen können. Die Windows Workflow Foundation ist als Teil des .NET Framework 3.0 auch kostenlos, jedoch ist ohne den Einsatz von anderen (lizenzpflichtigen) Microsoft Office Tools oder Servern keine direkte Verwendung möglich. Die Umsetzung von grundlegenden Workflow-Anwendungen ist zwar denkbar einfach, aber dafür sind Eigenimplementierungen unumgänglich. Derzeit wird der Windows Workflow Foundation keine große Bedeutung im Bereich Workflow Management und BPM zugeschrieben, mit Ausnahme im direkten Microsoft-Umfeld.

Sharepoint und BizTalk Server nur für die Abbildung von Geschäftsprozessen zu

verwenden, erscheint gemessen am gesamten Funktionsumfang der Systeme und der Lizenzkosten unverhältnismäßig. Ist jedoch eines der Systeme schon im Einsatz, bietet es sich an, es auch für den Bereich der Geschäftsprozessverwaltung einzusetzen.

Im Fall von schon vorhandenen Portalstrukturen im Unternehmen wird die Möglichkeit der Kombination von Portalssystemen mit Workflow Engines wie jBPM interessanter. So enthält zum Beispiel das Open Source Portal Liferay seit einiger Zeit ein Modul für die Verarbeitung von Workflows im Portal. Sicher ist Liferay mit seinem einzigen Workflow-Modul kein Prozessportal im klassischen Sinne, aber der erste Schritt hin zur Integration zweier Systemen ist gemacht und kann weiter ausgebaut werden.

Direkte XML-Definition oder „Zeichnen“ eines Prozesses?

Auch die Verfügbarkeit von Entwicklungstools für Workflows und Prozesse sollte in die Entscheidung für ein BPMS einfließen, falls der Prozess nicht direkt über eine XML-Notation durch die IT-Abteilung umgesetzt wird. Hier gibt es verschiedene Tools: Intalio bietet beispielsweise Plugins

XPD L	XML Process Definition Language, Definition durch WfMC, 2001 & 2004
WPDL	Vorgänger von XPD L, 1999
BPMN	Business Process Modeling Notation, Definition durch BPMI, 2004: Grafische Notation zur Beschreibung von Geschäftsprozessen, ist mittlerweile zum De facto-Standard geworden.
BPML	Business Process Management Language, Definition durch BPMI, 2001: XML-basierte Geschäftsprozessbeschreibung.
YAWL	Yet another Workflow Language, definiert durch die Queensland University of Technology in Australien in Zusammenarbeit mit Prof. Wil van der Aalst, u.a. bekannt durch das Benchmarken von WMS und Prozessbeschreibungssprachen mittels Workflow Pattern.
WSDL	Web Services Description Language, Definition durch OASIS: Generische XML-Sprache zur Beschreibung von Schnittstellen und Protokollbindungen der Bereitstellungsmöglichkeiten von Netzwerkdiensten. WSDL ergänzt den UDDI-Standard durch Bereitstellung eines einheitlichen Weges zur Beschreibung der abstrakten Schnittstellen und Protokollbindungen beliebiger Netzwerkdienste.
WSFL	Web Services Flow Language, IBM: Eine XML-Sprache zur Beschreibung des Aufbaus von Web Services als Teil einer Business Process Definition. Sie wurde von IBM als Teil des Web Service Technology Framework entwickelt und greift dabei auf bestehende Standards wie z.B. SOAP, WSDL, XMLP und UDDI zurück.
BPEL / BPEL4WS	XML-basierte Sprache für standardisierte Geschäftsprozesse in verteilten oder Clustersystemen, die es verschiedenen Geschäftszweigen erlaubt, ihre Anwendungen und Daten gemeinsam zu verwenden. Entwickelt als eine Kombination von IBMs WebServices Flow Language und Microsofts XLANG
XLANG	Siehe BPEL
WS-BPEL Extension for People	BPEL4People wurde so definiert, dass sie auf der BPEL-Sprache aufsetzt, um ihre Erweiterungen jederzeit mit BPEL-Kernfunktionen kombinieren zu können.

Bild 1: Überblick über Systeme und Prozessdefinitionssprachen.

Merkmal	XPDL	BPEL
Zielsetzung	Offener Austausch von Prozessdefinitionen (WfMC Interface 1) und Dateiformat für BPMN	Spezifikation von ausführbaren und abstrakten Geschäftsprozessen auf Basis von Web Services
Prozessteilnehmer	„Participants“: Ressource, Rolle, Org. Unit, Human, System Dynamische Zuordnung mittels Rules während Laufzeit	Intern keine Zuordnung von Teilnehmern, Externe im Message Flow über „Partner (Links)“
Technische Schnittstellen	Diverse, etwa Web Service, EJB, Pojo, Script, Rule	Web Services (WSDL) Binding beliebig
Manuelle Tasks	Explizite Definition von Aktivitäten als „TaskManual“	Nicht explizit unterstützt
Modularisierung	Unterprozesse werden unterstützt	Eingeschränkt möglich durch „scope“
Simulation	Enthält Attribute für Time Estimation, Cost Unit etc.	Nicht explizit unterstützt
Datenfluss	Nur bei Übergabe während Start und Ende eines Subprozesses	Kollaboration zwischen parallelen Web Services
Laufzeitverhalten, Problembehandlung	Für langlaufende automatische Aktivitäten (Teilprozesse) keine expliziten Definitionen	Enthält „fault handler“, „compensation handler“ und „transaction demarcation“
Austausch mit BPMN	Die Version 2.0 enthält neben Koordinaten- und Shape-Größen die Elemente Pool, Lane, Gateway und Event, also bidirektionaler Austausch mit BPMN möglich	Keine grafischen Elemente, nur unidirektionaler Austausch von BPMN nach BPEL möglich

Bild 2: Vergleich zwischen den zwei weit verbreiteten Sprachdefinitionen XPDL und BPEL.

für Eclipse, die es ermöglichen, alle Prozesse inklusive Ereignissen, Benachrichtigungen etc. grafisch zu definieren und zu parametrisieren. JBoss hat ein ähnliches Programm, den jBPM Designer als Eclipse Plugin, mit dem sich ganze jBPM-Projekte bearbeiten lassen. Microsoft setzt klassisch auf das Visual Studio 2005 und bietet als freie Erweiterung einen Prozessdesigner für die Windows Workflow Foundation an, der im Vergleich zu anderen Tools sehr gut abschneidet. Mit dem Import-Tool von BizTalk für Visual Studio können BPEL-Definitionen eingelesen werden. In Visual Studio 2007 (Orcas) wird der Process Designer wohl ein fester Bestandteil sein.

In der Regel bieten alle diese Anwendungen auch die Möglichkeit zum direkten Editieren der zugrundeliegenden Dateien – meist XML – inklusive der entsprechenden Syntaxhervorhebung.

Nachdem die Technik abgestimmt ist und die Entscheidung für die Entwicklungstools getroffen wurde, folgt die „Digitalisierung“ des Zielprozesses. Dabei werden mögliche Schwächen und Unwägbarkeiten des ausgewählten BPMS schnell deutlich werden. Das sollte einen jedoch nicht abschrecken. Hier soll allerdings auch nicht unterschlagen werden, dass ein BPMS

zwar große Fortschritte bringen kann, bei seiner Implementierung jedoch das eine oder andere Problem in Bezug auf den abzubildenden Prozess in Kauf genommen werden muss.

Wie sieht nun so ein „digitalisierter“ Prozess zum Beispiel in jPDL aus? Folgender Prozess aus Bild 4 soll beschrieben

werden. Das Beispiel zeigt eine grundlegende Struktur mit einer Verzweigung, einem Start- und einem Endpunkt, sowie den möglichen Übergängen zwischen den einzelnen Stati. Man sieht jedoch, dass das Beispiel noch keinerlei Aktionen enthält. Hier wird deutlich, dass ein an sich schon komplexer Prozess durch die Übertragung

WebSphere Process Server	Auf dem WebSphere AppServer basierender BPEL v1.1-kompatibler BPM Server
Oracle BPEL Process Manager	BPEL v1.1-kompatibel mit der Option auf eigene Entwicklung von „Human Tasks“
Microsoft BizTalk Server 2006	Verwendete Sprache ist XLANG, aber Import von BPEL-Definitionen über Visual Studio möglich
JBoss jBPM	BPEL-Support nur als Task in der Roadmap, jPDL wird für Prozessdefinition verwendet. Prinzipiell nur ein Framework.
Intalio BPMS	Unterstützt BPMN und BPEL, grafische Editoren zur Prozessbearbeitung vorhanden.
Microsoft Windows Workflow Foundation	Teil des .NET Framework 3.0, Workflow-Beschreibung via Code, Visual Studio Designer oder XAML-Definition möglich.
Office Sharepoint Server 2007	Der erste auf der Windows Workflow Foundation aufsetzende (kommerzielle) Workflow Server.

Bild 3: Einige aktuell verfügbare BPMS sowie einige Workflow Management Systeme/Engines, die zur Interpretation von Prozesssprachen eingesetzt werden können und rudimentäre Funktionen bieten.

in ein BPMS nicht einfacher wird – im Gegenteil. BPMS erfordern in der Regel immer einen vollständigen Prozess. Es sollte keinerlei „Sackgassen“ geben beziehungsweise Zustände, die nicht mehr verlassen werden können und keine Endpunkte sind (Bild 5).

Sollte es sich bei den Aktionen im System um reine Aufgabenverwaltung, Ein-/Ausgabe, Benachrichtigung und Sichtprüfungen handeln, ist die Abbildung des Prozesses im BPMS normalerweise einfach und ohne einen aufwändigen Einsatz von IT möglich. Der Prozessplaner kann daher sofort die Umsetzung seines Prozesses selbst in die Hand nehmen.

Auch Basislogik beispielsweise in Form der Interpretation von im Prozess erhaltenen Daten ist möglich, um etwa Verzweigungen auszulösen. Intalio bietet hier sehr viele Schnittstellen, die sich einfach parametrisieren lassen, sodass keinerlei Implementierungsaufwand entsteht. So-

bald allerdings weiterführende logische Operationen nötig sind, ist die IT gefordert, eigene Aktionen zu implementieren. Dafür bieten die verschiedenen Sprachdefinitionen und BPMS Möglichkeiten und Schnittstellen. Große Prozesse sind ohne solche Eigenimplementierungen kaum umzusetzen, weil die zu Grunde liegende Geschäftslogik viel zu komplex ist und sich im Gegenzug ein BPMS, das genügend konfigurierbare Logik enthält, kaum sinnvoll und einfach einsetzen lassen würde.

Fazit

Aktuelle Entwicklungen, speziell im Bereich BPMS, liefern wichtige Beiträge für den Aufbau des Geschäftsprozessmanagements, was zum gegenwärtigen Zeitpunkt allerdings nicht bedeutet, dass ein Prozess auf einfache Weise schon vollständig abgebildet werden kann. Unternehmen, für die Prozessmanagement nicht nur aus unterstützenden Teillösungen besteht, sondern

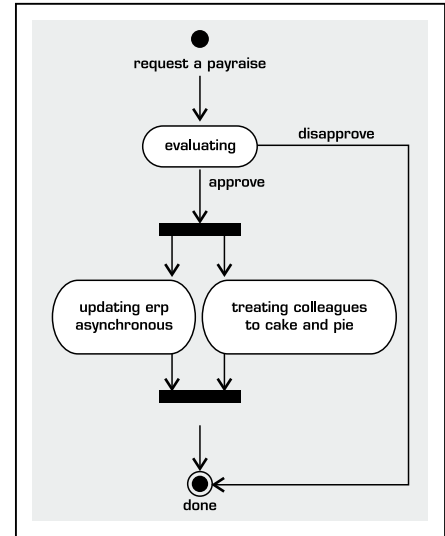


Bild 4: Dieser Prozess wird in Bild 5 beschrieben.

die komplexe Sachverhalte vollständig und durchgehend abbilden möchten, kommen nach wie vor nicht ohne Eigenimplementierungen aus.

Da für ein Unternehmen unabhängig vom jeweiligen Projekt immer Kosten- und Zeitaufwände anfallen (ob durch Eigenentwicklungen oder nur durch die Prozessumstellung), sollten diese Aufwände projektbezogen sein, um den Wert der Umstellung auch messen zu können. Business Process Management „um seiner selbst willen“ nützt keinem Unternehmen, egal, wie einfach die Umstellung auch sein mag.

Die Standardunterstützung der Hersteller und die inzwischen sehr weit gereiften BPM-Produkte, seien sie Closed oder Open Source, sowie ihre einfache Integration in bestehende IT-Landschaften werden dazu führen, dass immer weniger Unternehmen auf reine Eigenentwicklungen setzen, sondern auf BPMS zurückgreifen. Daneben ist zu erwarten, dass sich sowohl große Hersteller wie IBM, Oracle oder SAP als auch Open Source-Anbieter wie JBoss oder Intalio auf einen gemeinsamen Standard einigen werden. Ähnliches gilt für die aktuellen Sprachdefinitionen UML 2, BPMN, BPEL, XPD. Spannend bleibt, in welcher Richtung sich die Client-Plattformen konsolidieren werden – JSF/AJAX, SWT oder Microsoft – und wie sich hier Ansätze zur client-seitigen Anwendungsintegration etablieren.

Stefan Ehrlinger
Stefan.Ehrlinger@ancud.de

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE process-definition PUBLIC
  "-//jBpm/jBpm Mapping DTD 2.0//EN" "http://jbpm.org/dtd/processdefinition-2.0.dtd">

<process-definition name="pay raise process">
  ...
  <!-- START-STATE -->
  <start-state name="request a payraise" ...>
    <transition to="evaluating"/>
  </start-state>
  <!-- NODES -->
  <state name="evaluating">
    ...
    <transition name="approve" to="fork"/>
    <transition name="disapprove" to="done"/>
  </state>
  <fork name="fork">
    <transition to="updating erp asynchronously" />
    <transition to="treating colleagues on cake and pie" />
  </fork>
  <state name="updating erp asynchronously">
    ...
    <transition to="join" />
  </state>
  <state name="treating colleagues on cake and pie">
    ...
    <transition to="join" />
  </state>
  <join name="join">
  </join>
  <state name="done">
  </state>
  </process-definition>
  
```

Bild 5: Beschreibung des Prozesses aus Bild 4 in jPDL.

Security Eye
1/1 Seite

Security Eye
1/1 Seite