



>>> LIFECYCLE MANAGEMENT

Mit Prozessen den ALM-Workflow steuern

Individuell an die Herausforderungen im Unternehmen angepasst, schafft der Brückenschlag zwischen ALM- (Application Lifecycle Management) und Prozesswelt eine Arbeitserleichterung, ohne die Prozesskonformität zu verlieren. Er unterstützt die tägliche Arbeit im ALM-Tool und nimmt so auch unerfahrene Nutzer an die Hand. Aus einer unübersichtlichen Tool- und Prozesslandschaft entsteht ein Unternehmensprozess aus einem Guss.

Prozesse helfen dabei strukturiert zu arbeiten und die Anforderungen von Standards zu erfüllen. Application Lifecycle Management (ALM)-Tools erleichtern die Arbeit in den Teams. Prozesse und ALM-Tools sind meist in loser Kopplung miteinander verbunden und nicht aufeinander abgestimmt. Hier gilt es, eine Brücke zwischen Prozess und ALM-Welt zu schlagen, um prozesskonforme, automatisierte ALM Workflows zu erstellen.

ALM und Prozess

ALM-Tools, wie das von Atlassian entwickelte JIRA oder Rational Team Concert von IBM, haben in den meisten Unternehmen einen festen Platz. Sie erfüllen eine Vielzahl von Aufgaben und sind oft mit weiteren Tools vernetzt.

Ein ALM-Tool begleitet die Entwicklung einer Software durch alle Phasen des Softwarelebenszyklus. Vom Erfassen der Anforderungen, über die Entwicklung der Softwarekomponenten bis hin zu ihrer Umsetzung. Ziel des ALM-Tools ist die technische Unterstützung des Unternehmensprozesses. Von der ersten Idee bis zum Betrieb.

ALM-Tools sind als Helfer in der täglichen Arbeit konzipiert und bilden nur selten den ganzen Unternehmensprozess ab. Hier sind Tools, die sich auf Prozessmodellierung und Planung spezialisiert haben, klar im Vorteil.

Prozessmanagement-Tools helfen dabei Anforderungen von Standards wie SPICE und CMMI zu erfüllen und führen Unternehmen sicher durch Audits. Prozessmanager passen komplexe Prozessmodelle im Tool an das Unternehmen an und visualisieren diese. Für den Anwender sind die verschiedenen Prozessschritte verständlich dargestellt, sodass er den Prozess kontinuierlich weiterentwickeln kann.

Die meisten Unternehmen setzen Tools sowohl für das Application Lifecycle Management als auch für das Prozessmanagement ein, jedoch ohne die Schnittstellen klar zu definieren. So entsteht die Frage nach der Verknüpfung beider Welten meist erst während der Einführung dieser Tools. Die Lücke zwischen Prozess- und ALM-Tool wird nur Stück für Stück geschlossen und ohne Vision entwickelt. Dies meistern oft Einzelpersonen, zum Beispiel in der Rolle des Projektmanagers. Kleine Eigenentwicklungen, vom Excel-Import bis hin zu komplett eigenständigen Anwendungen, unterstützen sie dabei.

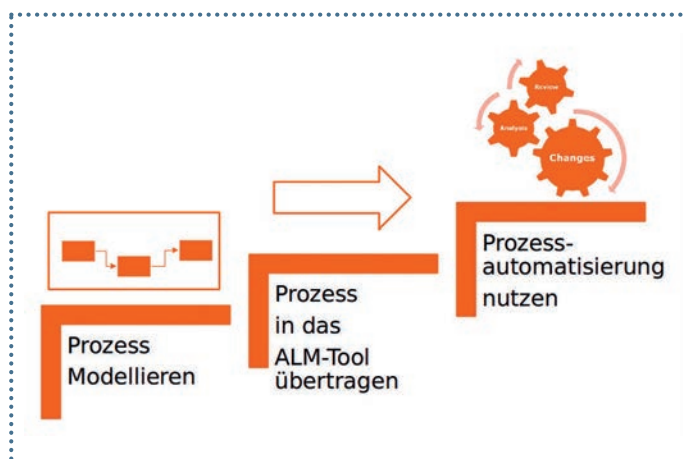


Bild 1: Der modellierte Prozess wird in das ALM-Tool übertragen und hat somit direkt Einfluss auf die Prozessautomatisierung.

(© Method Park)



Der Projektmanager ist dann einen Teil seiner Zeit damit beschäftigt, dem Team Aufgaben prozessgerecht zuzuschneiden und diese im ALM-Tool festzuhalten. Eigenentwicklungen schmälern die Lücke, schaffen aber oft auch neue Probleme. Sie müssen auf dem neusten Stand gehalten werden, um mit den Tools kompatibel zu bleiben, die sie unterstützen. Anwender benötigen ausreichend Dokumentation, damit mehr als eine Person die Erweiterungen bedienen kann. Durch eine Vielzahl von Eigenentwicklungen entsteht eine unübersichtliche Tool-Landschaft. All das prozesskonform unter einen Hut zu bringen ist eine Sisyphusarbeit. So wird Arbeitskraft in Infrastruktur gebunden und ein gelebter Unternehmensprozess geschaffen, der an keiner Stelle komplett festgehalten oder geplant wird.

Sobald Projekte eine gewisse Komplexität erreichen, gibt es kaum noch Personen im Unternehmen, die den Überblick behalten. Der Besuch des Auditors wird zur Hängepartie, und mit der Zeit schleichen sich mehr und mehr Provisorien ein. Das Bindeglied zwischen ALM- und Prozess-Tool fehlt. Das produziert unnötige Arbeit, und Fehlerquellen entstehen.

ALM und Prozesswelt

Um diesen Spagat zu meistern und die Lücke zwischen ALM- und Prozess-Tool zu schließen, gibt es die Möglichkeit ALM-Tools mit Prozesselementen zu befüllen sowie den Workflow zu konfigurieren und zu automatisieren. Für diese Konfiguration müssen Prozessschritte und Arbeitsprodukte aus dem Prozess in Workflows und Aufgaben übersetzt werden.

Ein ALM-Tool arbeitet im Allgemeinen mit Aufgaben in verschiedenen Status. In der Prozesswelt entsprechen diese den Arbeitsprodukten. Aktivitäten im Prozess repräsentieren den aus Statusübergängen bestehenden Workflow.

Wenn ein Entwickler einen Bug im ALM-Tool vom Status „New“ zu „In Progress“ ändert, beginnt er aus Prozesssicht mit der Aktivität „Fehler beseitigen“. Die Aktivität „Fehler dokumentieren“, die als Arbeitsprodukt den Bug im ALM-Tool vorschreibt, deckt wiederum das Anlegen des Bugs ab.

```
var owner = .getCurrentUser();

var task = createWorkItem("Project A", "task", "Aufgabe 1");
task.setOwner(owner);

var taskAction = WorkflowsLib.getActionForResultState(TASK_A, "In Progress");
task = commitWorkItemAndChangeState(TASK_A, taskAction, true);
```

Programmcode 1: Mit wenig Aufwand lässt sich eine Aufgabe erstellen, sie „In Progress“ setzen und abhängig vom jeweiligen Nutzer einem Verantwortlichen zuweisen. (© MethodPark)

```
var creatorWorkItem = getCurrentWorkItem();
var endPoints = [ EndPoints.RESOLVED_BY_WORK_ITEM ];
var stagesActivityElement = getAssociatedElement();

StagesWorkflowLib.createChildrenAndReferences(creatorWorkItem,
stagesActivityElement, endPoints, true);
```

Programmcode 2: Soll am Anfang eines Prozessschrittes aus jeder „Story“ ein bestimmtes Set an Aufgaben generiert werden, kann das einfach automatisiert werden. (© MethodPark)

So ist es möglich den kompletten Workflow im Unternehmensprozess einzubetten und das ALM-Tool mit Prozesskontext anzureichern. Diese Verzahnung ermöglicht es komplett nachvollziehbar zu arbeiten. Die Eigenentwicklungen werden obsolet und durch Teile des Prozesses ersetzt.

Einbindung von ALM-Tools

Bei der Verzahnung von ALM- und Prozess-Tool übernimmt das Prozess-Tool die beschreibende Rolle. Es definiert die Arbeitsabläufe unabhängig vom verwendeten ALM-Tool. Welche Statusübergänge im ALM-Tool unter welchen Bedingungen geschehen und welche Aktionen ausgeführt werden können, ist dem jeweiligen Projektmanager überlassen und nicht an den Prozess gekoppelt.

Der Prozess gibt zum Beispiel vor, dass ein weiterer Entwickler jede Programmieraufgabe überprüfen muss. Das kann bedeuten, dass der Entwickler, der die Aufgabe bearbeitet hat, nicht derselbe Entwickler sein darf, der diese Aufgabe abschließt. Es kann aber auch bedeuten, dass nach Beendigung der Aufgabe eine weitere Überprüfungsaufgabe angelegt wird, die wiederum von einem anderen Entwickler abgearbeitet

wird. Beides wird händisch im ALM-Tool durchgeführt.

Will man einheitlich arbeiten, kann man den Prozess so gestalten, dass er die Ausführungsdetails schon beinhaltet und so sowohl die Entscheidung über die Ausführung als auch die Schritte der Ausführung übernimmt.

Dies geschieht in der Regel durch detailliertere Prozesse, die mit Ausführungsskripten angereichert sind. Der Prozess beinhaltet sowohl ALM-spezifische als auch unabhängige Prozessschritte. Die ALM-spezifischen Schritte können mit Skripten angereichert werden, die das ALM-Tool bis ins kleinste Detail steuern. Alles natürlich abhängig vom Prozess.

Technisch wird der Prozess, inklusive der Skripte, über einen Export an das ALM-Tool übertragen. Das garantiert, dass zu einem definierten Zeitpunkt ein definierter Prozessstand und die damit verbundenen Automatisierungen im ALM-Tool bestehen. So steht dem ALM-Tool der gewünschte Prozesskontext jederzeit zur Verfügung und kann im Prozess-Tool weiterentwickelt werden.

Das ALM-Tool ist nun in der Lage, abhängig vom übergebenen Prozess, auf Nutzeraktionen zu reagieren. In der Konfiguration des ALM-Tools sind alle prozessrelevanten Informationen hinterlegt. »

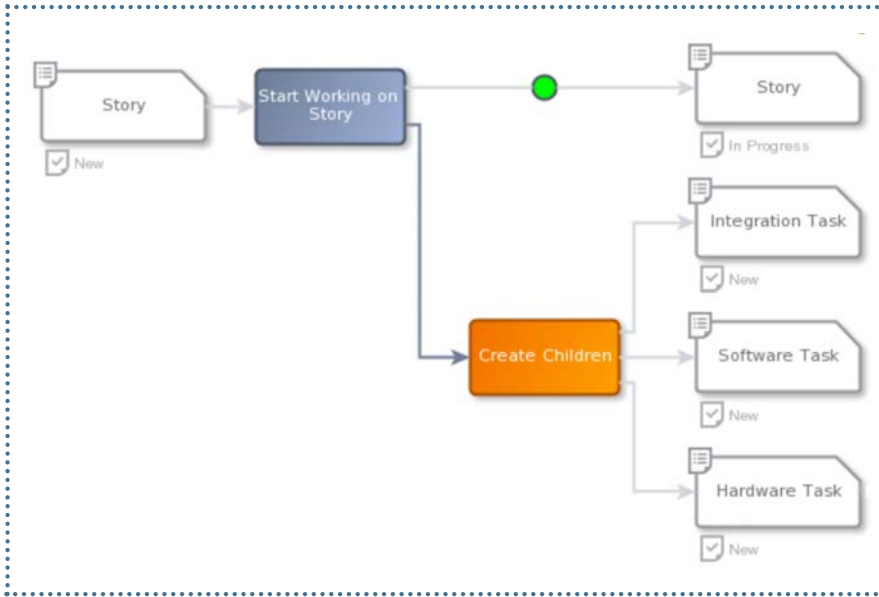


Bild 2: Prozessmodell für eine Automatisierung zum Anlegen von Unteraufgaben.

© Method Park

Anreicherung mit Skripten

Die Automatisierungen werden durch das Prozessmodell beschrieben und in JavaScript realisiert. Damit ist die volle Flexibilität einer Programmiersprache gegeben und auch komplexe Sachverhalte können abgebildet werden. Das Prozessmodell ermöglicht eine einfache Darstellung der Automatisierung.

Vom Ändern von Attributen bis zum kompletten Anlegen eines Projekts und seiner Teilaufgaben ist alles möglich – sowohl initial als auch während des laufenden Projekts. Die Automatisierungen können flexibel auf die aktuellen Gegebenheiten reagieren.

Die Skripte sprechen direkt die APIs der verschiedenen ALM-Tools an. So sind auch Aktionen möglich, die nur über komplexe Einstellungen erreicht werden können oder dem Nutzer komplett verwehrt bleiben. Vor allem Prozesse, die oft wiederholt werden oder eine Menge an Aufgaben betreffen, können so sehr effektiv ausgeführt werden.

Auch die Anbindung weiterer Systeme ist über die Skripte möglich. So ist es einfach, aus den Skripten REST Services anzusprechen und zusätzliche Daten in die ALM-Welt zu integrieren oder diese an andere Systeme zu übertragen. Alles immer im Kontext des Prozesses.

Beschreibt der Prozess zum Beispiel, dass jede Aufgabe und jedes Do-

kument einen Verantwortlichen haben muss und dieser initial der Ersteller der Aufgabe ist, dann reichen ein paar Zeilen aus, um eine Aufgabe zu erstellen, sie „In Progress“ zu setzen und die Aufgabe abhängig vom jeweiligen Nutzer einem Verantwortlichen zuzuweisen (Programmcode 1).

Ausführung der Automatisierungen

Die Skripte werden beim Speichern einer Aufgabe im ALM-Tool ausgeführt und reagieren auf den im Modell beschriebenen Statusübergang. So gibt es zwei Möglichkeiten in den Workflow einzugreifen: wenn ein Statuswechsel vorgenommen oder wenn eine Aufgabe in einem bestimmten Status gespeichert wird. Der Prozessmanager modelliert im Prozess die Randbedingungen. Ein Entwickler programmiert im Script die konkrete Aktion für das ALM-Tool. Mithilfe des Prozesses lässt sich das ALM-Tool einfach konfigurieren.

Soll zum Beispiel am Anfang eines Prozessschrittes aus jeder „Story“ ein bestimmtes Set an Aufgaben generiert werden, kann das, wie in Bild 2 (Transition Workflow Create Children) modelliert, mit dem im Programmcode 2 gezeigten automatisiert werden.

Das Skript legt beim Speichern der Story alle im Modell definierten Aufgaben an (im Beispiel eine Software, eine Hardware sowie eine Integration Task)

und sorgt dafür, dass sie die Attribute aus dem Modell übernehmen. Das geht sogar so weit, dass komplette Templates für ALM-Aufgaben im Prozess vordefiniert sein können.

Wenn ALM-Elemente über die Prozessautomatisierung erzeugt wurden, erhalten sie ihren speziellen Prozesskontext. So sind sowohl Hardware-, Software- als auch Integration-Task Tasks, wie sie das ALM-Tool definiert. Ihnen gemeinsam sind die Eigenschaften von Tasks; sie können diese aber um ihren speziellen Prozesskontext erweitern.

So könnte zum Beispiel die Integration-Task eine besondere Eigenschaft haben. Sie kann nur geschlossen werden, wenn die Software in die Hardware integriert ist. Also müssen, bevor eine Integration-Task abgeschlossen werden kann, die zugehörigen Software- und Hardware-Tasks geschlossen sein. Trotz dieser speziellen Eigenschaft teilt die Integration-Task aber immer noch eine Bedingung mit allen anderen Tasks: Es muss einen Verantwortlichen geben. Beides ist über den Prozess festgelegt.

Fazit

Im Einsatz bei Großprojekten mit mehreren hundert Entwicklern zeigt die Automatisierung ihre ganze Stärke. Wo im Einzelnen ein paar Minuten gespart werden, werden im Großen und Ganzen Tage gespart. Und das ohne den Verlust von Struktur oder Prozessstreue. Was vorher zeitraubende Handarbeit war, geschieht durch die Automatisierung in Sekundenbruchteilen im Hintergrund.

Individuell an die Herausforderungen im Unternehmen angepasst schafft der Brückenschlag zwischen ALM- und Prozesswelt eine Arbeitserleichterung, ohne die Prozesskonformität zu verlieren. Er unterstützt die tägliche Arbeit im ALM-Tool und nimmt so auch unerfahrene Nutzer an die Hand. Aus einer unübersichtlichen Tool- und Prozesslandschaft entsteht ein Unternehmensprozess aus einem Guss. ■ (oe)

» www.methodpark.de



Peter Neubauer ist Senior Solution Architect bei Method Park und zertifizierter Scrum Master.