

Flexibel und modular: Service-orientierte Architekturen für das Requirements Management

Martin Schilke, Bexbach

Produzierende Unternehmen stehen heute einer Reihe von Veränderungen gegenüber. Die Funktionalität und Komplexität der Produkte nimmt zu. Damit verbunden steigt nicht nur die Anzahl der Bauteile, sondern auch der Anteil an elektrischen bzw. elektronischen Komponenten (sogenannten „E/E-Komponenten“) und Software in ehemals vorwiegend mechanisch geprägten Produkten. Für die Entwicklung solcher mechatronischen Produkte wird die Zusammenarbeit zwischen den einzelnen Fachdisziplinen Mechanik, Elektronik und Informationstechnik immer wichtiger. Darüber hinaus sind Unternehmen heute in der Situation, nicht mehr nur Komponenten zu liefern, sondern als Lieferant für ein funktionierendes Gesamtsystem aufzutreten. Die Vernetzung und die Abhängigkeiten der mechanischen und E/E-Komponenten untereinander lässt eine getrennte Betrachtung und Entwicklung dieser Komponenten nicht zu und tragen zusätzlich dazu bei, dass sich die Komplexität des Gesamtsystems erhöht.

Um diese Herausforderungen zu bewältigen und erfolgreiche Produkte auf den Markt bringen zu können, beschäftigen sich produzierende Unternehmen verstärkt mit dem Ansatz des *Systems Engineering*. Hier geht es darum, ausgehend von den Anforderungen an ein System die erforderlichen Funktionen zu definieren und diese in Produktkomponenten umzusetzen. Die Rückmeldung von Testergebnissen und der Erfüllungsgrad von Anforderungen ist ein wesentlicher Bestandteil des Systems Engineering Prozesses.

Ausgangspunkt für die Systementwicklung sind die Kundenanforderungen, die auf Systemebene formuliert, im Laufe

des Entwicklungsprozess verfeinert und schließlich auf die Disziplinen Mechanik, E/E und Software verteilt werden. Durch die Festlegung von Funktionen und deren weitere Aufteilung in Teilfunktionen ergibt sich eine Funktionsstruktur. Im Zuge der Umsetzung der Funktionen baut sich analog die Produktstruktur auf. Durch Verknüpfung von Anforderungs-, funktionaler und Produktstruktur ergibt sich für das System ein mitunter komplexes Beziehungsnetzwerk, wie Abbildung 1 zeigt. Eine Aufgabe des Systems Engineering ist es in diesem Zusammenhang, die Konsistenz der einzelnen Strukturen zu gewährleisten, um beispielsweise die Auswirkungen einer Anforderungsänderung auf die verknüpften Objekte der

verschiedenen Disziplinen transparent zu machen.

Betrachtet man die informationstechnische Systemlandschaft in der Produktentwicklung, so kommen sowohl innerhalb einer Fachdisziplin wie auch zwischen den verschiedenen Disziplinen unterschiedliche, auf den konkreten Anwendungsfall ausgerichtete Werkzeuge wie PDM, CAD, CAE, CAM oder CASE zum Einsatz. Die Integration der beteiligten Systeme ist notwendig, um ein gesamtheitliches Requirements Management realisieren zu können. Einen Ansatz hierzu stellen sogenannte *Service-orientierte IT-Architekturen* (engl. *service-oriented architecture, SOA*) dar.

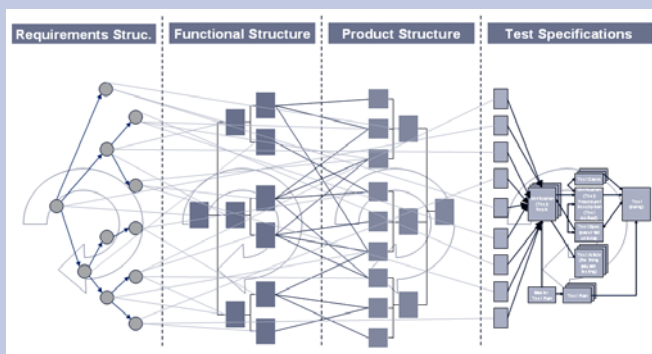


Bild 1: Datenstrukturen in der Produktentstehung

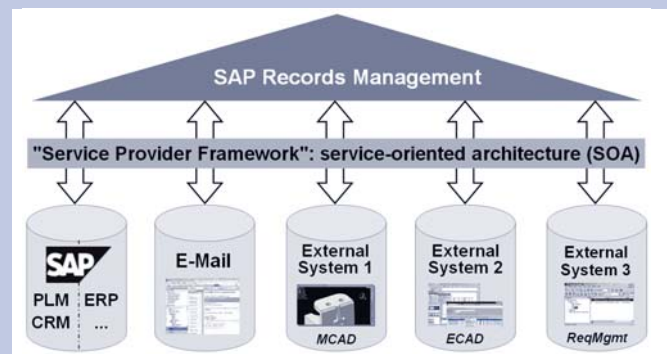


Bild 2: Service-orientierte Architekturen mit SAP Records Management

Service-orientierte Architekturen sind flexible und modulare IT-Architekturen, die in der Lage sind, Daten aus verschiedenen und verteilten Quellen integriert bereitzustellen. Die Abbildung von Methoden und der Zugriff auf die am Prozess beteiligten Applikationen werden über Dienste (Services) realisiert, die über standardisierte Schnittstellen in eine zentrale Plattform integriert werden können. Eine technologische Basis zur Implementierung solcher Service-orientierten Architekturen bildet das SAP Records Management, mit dessen *Service Provider Framework* sich bereits im Standard vorhandene SAP-R/3-Systeme integrieren und darüber hinaus Integrationen für weitere Drittsysteme (Legacy Systems) wie z.B. Requirements Management Systeme entwickeln lassen. Service Provider realisieren das Einbinden von Informationsobjekten in SAP Records Management und regeln den Zugriff darauf. Somit kann auf alle für einen Unternehmensprozess notwendigen Informationen von einem zentralen Punkt aus zugegriffen werden. Abbildung 2 gibt einen Überblick über die Systemarchitektur des SAP Records Managements.

Durch diese Integrationsmöglichkeiten bietet SAP Records Management eine durchgängige und applikationsübergreifende Sicht auf alle zum Produktentstehungsprozess gehörenden Informationsobjekte. Der Zugriff auf Informationen wird insofern erleichtert, da der Anwender nicht mehr in den jeweiligen IT-Systemen Informationsobjekte suchen muss, sondern alle Informationsobjekte in einer für ihn strukturierten Sicht findet und von dort aus direkt in das Zielsystem navigieren kann, um dort die Objekte zu bearbeiten.

ILC PROSTEP hat in Zusammenarbeit mit einem namhaften deutschen Automobil-

zulieferunternehmen eine Systems Engineering Plattform auf Basis des SAP Records Management aufgebaut, die auch Aspekte des Projektmanagements berücksichtigt. Zur Verwaltung und Ablage der Produktdaten und zum Aufbau von Stücklisten kommt SAP PLM zum Einsatz, wodurch kein externes PDM-System benötigt wird. Die relevanten Informationsobjekte für die in Abbildung 1 dargestellte Produktstruktur wurden direkt als Services aus den angebotenen SAP-R/3-Systemen in die Systems Engineering Plattform integriert, ohne zusätzliche Schnittstellen einzusetzen. Für die Verwaltung und Bearbeitung der Anforderungen und Spezifikationen kommt das Requirements Management System MKS Integrity zum Einsatz. Die entstehenden Anforderungsstrukturen müssen in die Systems Engineering Plattform in SAP integriert werden, um sie mit den dort vorhandenen funktionalen und Produktstrukturen zu verknüpfen. Dazu wurde ein neuer Service Provider entwickelt, über den MKS Strukturen in SAP Records Management visualisiert und über bereit gestellte Methoden im MKS Integrity Client zur Anzeige oder zum Ändern geöffnet werden können. Ein Beispiel hierfür zeigt Abbildung 3, in der zu einer Funktion im Records Management die zugehörigen Anforderungen aus MKS Integrity und deren Verknüpfungen in der Struktur dargestellt sind. Ein direktes Abspringen in den MKS Integrity Client erfolgt über das Kontextmenü zu einer Anforderung.

Der Service Provider greift über eine Middleware auf den MKS Integrity Server zu. Dabei handelt es sich um die nach dem Industriestandard J2EE entwickelte Softwarelösung „Mechatronic Integration Suite“ der ILC PROSTEP, deren Architektur auf SAP NetWeaver basiert und im Rahmen des „POWERED BY SAP Net-

Weaver Programms“ der SAP zertifiziert ist (Abbildung 4). In einem Repository werden Funktionskomponenten (Activities) zum Bilden der Prozesse verwaltet und bereitgestellt. Beliebige Activities, Transformationen und Datentypen können in einem „Baukasten“ hinterlegt werden. In den Prozessen werden diese Einheiten dann zu einer Integrationslösung zusammengestellt. Die Funktionskomponenten und Prozesse der ILC Mechatronic Integration Suite sind vollständig via Web-Services in SAP-Umgebungen integrierbar.

Die positiven Erfahrungen mit der Systems Engineering Plattform im SAP Records Management zeigen die Potentiale der Service-orientierten Architektur. Nach der Einbindung des Requirements Management Systems MKS Integrity gilt es nun, noch weitere Applikationen entlang des Produktentstehungsprozesses über Service Provider zu integrieren. Zum einen geht es dabei um den Zugriff auf Testspezifikationen und -ergebnisse aus dem Versuchsmanagement, zum anderen um die Integration der funktionalen Modellierung. In der aktuellen Ausbaustufe sind die Funktionen als SAP Records Management Objekte abgebildet, während die eigentliche Modellierung der funktionalen Strukturen in einem SysML-Werkzeug erfolgt. Durch direkte Einbindung dieser Strukturen könnten auch die frühen Phasen der Produktentwicklung in der Systems Engineering Plattform verfügbar gemacht und deren Effizienz weiter gesteigert werden. ■

Kontakt

Martin Schilke
ILC PROSTEP GmbH
E-Mail: info@ilc-prostep.de

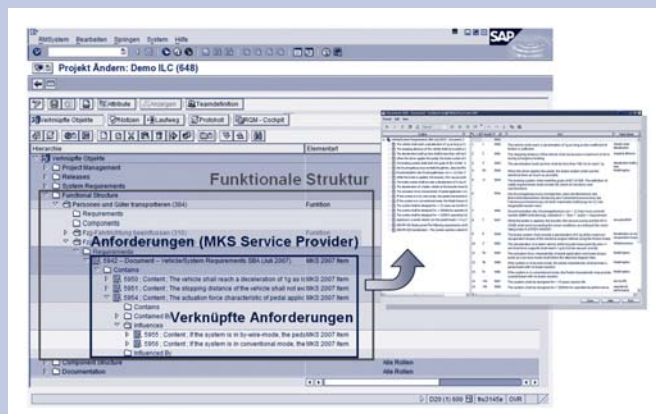


Bild 3: Systems Engineering Plattform im SAP Records Management

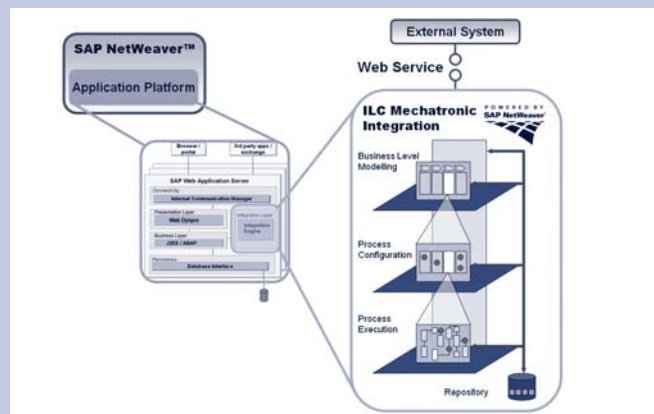


Bild 4: Architektur der ILC Mechatronic Integration Suite