

Wert von Offenheit, Teil II

Wir wissen nicht, was der junge Mann denkt; was wir allerdings wissen, ist, dass die Business-Strategen aufseiten der PLM-Systemanbieter und -anwender diese Freiheit nicht genießen – zumindest derzeit nicht. Kann der CPO ihnen helfen?

Langfristig Investitionen sichern

Im Jahr 2011 wurde auf Anregung von Dietmar Trippner, eines Vertreters der Anwenderseite, der Code of PLM Openness (CPO) aus der Taufe gehoben. Gemeinsam haben PLM-Systemanbieter, Systemintegratoren und Anwender ein Rahmenwerk geschaffen, das beschreibt, welche Eigenschaften der Offenheit aus den Interessen der unterschiedlichen Betroffenen heraus bei IT-Systemen und ihrer Implementierung erfüllt sein müssen, sollten oder könnten.

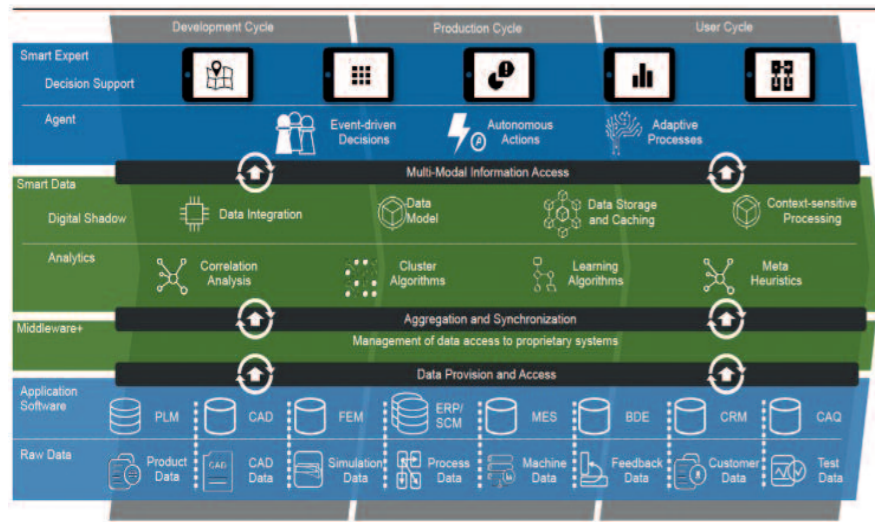
Im April 2017 erschien in dieser Zeitschrift an gleicher Stelle Trippners Beitrag „Wert der Offenheit“, der die Anwendersicht darlegt. DOMINIK RÜCHARDT stellt nun die Sicht eines Softwaresystemanbieters vor, der von Beginn an aktiv an der Gestaltung des CPO beteiligt war.

Die Geschichte der Softwareindustrie ist noch relativ jung. Beim Blick nur 30 Jahre zurück finden wir eine Welt der Lochkarten, Großrechner und seltsamen ersten Tischcomputer vor – und stoßen auf den Begriff „DOS compatible“, der über Erfolg oder Misserfolg der damals neuen PCs entschied.

Die Entwicklung seitdem ist beispiellos – und voller Überraschungen. Abgesehen von der Konstante „Moore’s Law“, das über Jahrzehnte hinweg verlässliche Aussagen über die zu erwartende Prozessorgeschwindigkeit und die Speicherkapazitäten machte, konnte zu keinem Zeitpunkt jemand voraussagen, was in fünf geschweige denn in zehn Jahren das beherrschende Thema oder die beherrschende

IT-Technologie sein würde. Standardisierungsbemühungen waren häufig schneller obsolet, als sie durchgesetzt waren, gewaltige Investitionen fanden statt, und Strategien litten nicht selten unter dem Problem, dass sich bis zu ihrer Umsetzung die Rahmenbedingungen bereits wieder komplett verändert hatten.

Besonders betroffen ist dabei das Gebiet der Unternehmenssoftware, allen voran PLM. Der Grund liegt in der Schlüsselrolle rund um die Produktentstehung, bei der Kreativität auf Umsetzung und Produktvielfalt auf hohe Änderungsdynamik trifft und viele Fachdisziplinen gleichermaßen involviert sind, mit all ihren unterschiedlichen Kulturen und Eigenheiten. Und mit zunehmender Digitali-



Internet of Production: Beispiel einer Integrationsarchitektur für Closed-Loop Lifecycle Management im vernetzten Unternehmen. Dieses Konzept erfordert drei wesentliche Zutaten:

- Einen Architekturplan, der die vernetzten Unternehmensprozesse unterstützt. Solch ein Plan sollte – soweit irgend möglich – standardisiert sein, sich also an branchenweite Best Practices anlehnen.
- Eine robuste Vernetzungstechnologie, die Systemintegration, einfache UI-Gestaltung und alle Aspekte einer digitalen Prozesskette wie Data Analytics, AR und Kopplung an Dinge und Maschinen im Betrieb (IIoT) unterstützt.
- Den Einsatz von Werkzeugen und Systemen, die in der Lage sind, sich in eine offene, stets weiterentwickelnde Systemumgebung einzufügen.

Quelle: WZL / Fraunhofer IPT

sierung bei Produkten, Prozessen sowie der Märkte insgesamt werden die Gräben zwischen den unterschiedlichen Auffassungen noch tiefer.

Die Geschichte der Software ist eine der Evolution, nur im Zeitraffer. Und damit ist bereits fast alles gesagt. Das Kerngesetz dieser Veränderung, von Charles Darwin trefflich formuliert, lautet: „Survival of the Fittest“ – das Überleben des Anpassungsfähigsten.

Protektionismus hat seinen Preis

Aus der Sicht eines Softwareherstellers bedeutet das: Man kann in kurzer Zeit stark wachsen, muss sich aber gleichzeitig darüber im Klaren sein, dass sich auch alles wieder sehr schnell ändern kann. Das prägt die Strategien, nach denen Softwarehersteller ihr Handeln bestimmen, wenn es darum geht, die eigene Position im Markt zu stärken. Im Prinzip sind es drei Möglichkeiten:

- Abhängigkeit – ein Systemwechsel wird so schmerzhaft und teuer gemacht, dass ihn keiner will
- Abschottung – mittels technischer Hürden wird Wettbewerbern der Marktzutritt versperrt
- Offenheit – man sorgt dafür, dass es unkompliziert ist, sich für die eigene Software zu entscheiden.

Dabei ist es mitnichten so, dass diese drei Strategien stets bewusst gewählt werden. Oft ergeben sie sich aus dem laufenden Geschäft, werden zunächst gar nicht als explizite Entscheidungen wahrgenommen und sorgen erst später für Erfolge oder Konflikte. Unabhängig davon gibt es bei jeder Strategie ein Für und Wider:

- Abhängigkeit – scheint zunächst attraktiv, wird aber zum Problem, sobald sich mittelfristig die Interessen von Kunden und Softwarehersteller unterschiedlich entwickeln. Langwierige, verschleißende Auseinandersetzungen sind die Folge, die in der Regel keinen Gewinner haben.
- Abschottung – wird in der Regel mit Qualitätssicherung begründet, funktioniert aber nur dann, wenn innerhalb

des abgeschotteten Bereiches tatsächlich ausreichend Gestaltungsraum ist, um die Bedürfnisse der Kunden zu bedienen. Wird zum Problem, je vernetzter Kunden und deren Aufgaben werden. Abschottung ist also eher dafür geeignet, einen etablierten Bestandmarkt über eine gewisse Zeit abzusichern.

- Offenheit – ist bei Kunden stets die beliebteste Form. Birgt das Risiko, dass die Treue der Kunden gelegentlich doch nicht so groß ist wie gedacht und diese zu einem günstigeren Anbieter wechseln, der sich auf das Nachimplementieren der Bestseller fokussiert. Weitere Gefahren sind der Missbrauch von Schnittstellen und das Schaffen von Abhängigkeiten, die notwendigen Innovationen im Weg stehen.

Die genannten drei Strategien sind also zu kurz gedacht, um den Markterfolg von Systemanbietern zu erklären.

Softwarehersteller müssen in erster Linie priorisieren und Strukturen vereinfachen

Im realen Geschäftsleben eines Softwareherstellers weichen die maßgebenden Faktoren erheblich davon ab: Die Kundenwünsche übersteigen die Möglichkeiten der Umsetzung, bei komplexen Themen kann es immer wieder zu Missverständnissen bei den Erwartungshaltungen entlang der Entscheidungskette kommen, und kurzfristige Eskalationen stören die Planung. In der betrieblichen Praxis entstehen ganz natürlich neue Bedarfe nach Unterstützung von Formaten, Systemen, Versionen oder Schnittstellen. Die Lösungen sind oft schnell entwickelt, die Kosten dafür, all das über Jahrzehnte bei allen Versionsänderungen durch die Qualitätssicherung zu tragen, sind aber enorm. Schließlich müssen alle Kombinationen jedes Mal aufs Neue überprüft und getestet werden. Die wichtigste Aufgabe von Produktmanagern ist daher die Priorisierung bei begrenzten Mitteln – im Sinne des Herstellers, der nicht unendlich viele Konzepte verfolgen kann, sowie im Sinne des Kunden, der klar verständliche und nachhaltige Konzepte erwartet.

Jenseits der drei zuvor genannten Strategien sind daher für Hersteller wie auch Kunden andere Aspekte wichtiger, um langfristig erfolgreich zu sein.

Unternehmenssoftware lebt von vertrauensvollen Beziehungen

Zuverlässigkeit. Langfristig gute Kundenbeziehungen sind von elementarer Bedeutung, um die tatsächlichen Bedürfnisse in einem Markt zu kennen und deren Abdeckung partnerschaftlich zu gestalten – und auch als solides wirtschaftliches Fundament. Auch wenn Softwareherstellern oft die vergleichsweise hohen Margen vorgeworfen werden, wird gerne übersehen: Der hohe Innovationsdruck im Softwaregeschäft ist teuer.

Leumund. Es gibt keine bessere Werbung als zufriedene Kunden. Gerade im Geschäft mit Unternehmenssoftware sind die kritischsten Themen Integration in die Unternehmensprozesse und Anwendungsumgebungen sowie partnerschaftliches Zusammenspiel. Darüber hinaus führt Misstrauen im Kundenverhältnis dazu, dass bei komple-

xen Themen wie Unternehmenssoftware ein zunehmendes Missverständnis zwischen Angebot und Nutzung entsteht – eine fatale Spirale.

Entwicklungspotenzial. Im innovationsgetriebenen Softwaregeschäft ist es dringend nötig, vorhandene Lösungen so zu gestalten, dass sie als robuste Basis auch weitere Innovation zulassen. Das betrifft zum einen die Architektur und einfache Erweiterbarkeit von Lösungen, zum anderen die verfügbaren Kapazitäten. Wenn alle Mitarbeiter damit beschäftigt sind, Konflikte und Probleme rund um die laufenden Lösungen zu bedienen, ist kein Platz für Innovation.

Diese drei Kriterien lassen schnell erkennen: Für Softwarehersteller, die sich aktiv weiterentwickeln und wachsen wollen, gibt es keine Alternative dazu, stabile, offene Architekturen anzubieten, die sich gut in die Umgebungen und Abläufe ihrer Kunden integrieren. Doch das ist auch aus Sicht eines Softwareherstellers leichter gesagt als getan, denn die Dynamik des Softwaregeschäftes steht einer langfristigen Planung häufig im Weg und verlangt Kompromisse.

Priorisierung der eigenen Kapazitäten. Die begrenzte Anzahl der Entwicklungsressourcen auf notwendige Innovationen oder auf Schnittstellen anzusetzen, ist ein ständiger Prioritätenkonflikt.

Bedienen vieler Interessen. Die einzelnen Kunden und Kundengruppen haben oft sehr verschiedene Vorstellungen. Alle bedingungslos zu bedienen ist weder möglich noch sinnvoll. Oft ist ein zähes Ringen zwischen dem Softwarehersteller und dem Kunden nötig, um auf langfristig sinnvolle Lösungen zu kommen.

Einschätzung von Trends. Angesichts des hohen Innovationsstemplos ist oft noch nicht zu erkennen, welcher Weg sich durchsetzen wird. Andererseits erzeugen einmal realisierte Lösungen einen enormen Wartungs- und Pflegeaufwand. Hersteller zögern daher oft mit der Umsetzung bestimmter Technologien, bis sie entscheidungssicher sind.

Ähnlich wie andere Unternehmen haben auch Softwarehersteller unterschiedliche Schwerpunkte in ihrem Geschäftsmodell. Die einen setzen einen Schwerpunkt auf Standardlösungen, die anderen auf Innovation, wieder andere auf das Projektgeschäft.

Nachhaltige Innovation braucht Partnerschaft

Unabhängig davon sind aus der Sicht eines Softwareherstellers die wichtigsten Eigenschaften von Offenheit:

Verständliche Architekturkonzepte – vorausgesetzt, dass sich sowohl Hersteller als auch Kunden und Serviceanbieter daran halten. Diese Konzepte sind natürlich dynamisch und entwickeln sich kontinuierlich weiter, getrieben durch Innovation und Wettbewerb. Dazu müssen sie von der Community aus Herstellern, Kunden, Beratern und Akademikern gepflegt werden. Das reicht von der Standardisierung auf Funktionsebene bis zu den großen Sollbruchstellen in einer Unternehmensarchitektur.

Aktives Rollenverständnis. Die Komplexität des Softwaregeschäftes wird von Kunden oft unterschätzt. Ebenso wird unterschätzt, in welchem Umfang sich auf Kunden-seite Prozesse und Organisation weiterentwickeln müssen, wenn innovative Softwarelösungen eingeführt werden.

Partnerschaftliche Zusammenarbeit. Unternehmenssoftware ist immer eine Sache von vielen Beteiligten und Prozessen, das erschwert viele Entscheidungsvorgänge. Das betrifft Gruppen in einem Unternehmen und das Verhältnis zwischen Kunden und Anbietern, Drittanbietern und Interessenvertretern von Branchen. Je besser sich diese Gruppen untereinander organisieren und abstimmen, desto besser werden die Ergebnisse – zum Nutzen aller Beteiligten.

Die Bedeutung dieser Aspekte von Offenheit nimmt weiter zu. Das liegt an der zunehmenden Komplexität vernetzter Systeme, die das Industriegeschäft ausmachen.

Zukünftige Konzepte bauen auf offene Systeme

Der gängige Ansatz für die Systemarchitektur um die Jahrtausendwende hat zu leistungsfähigen Systemwelten ge-

führt, deren weiterer Ausbau allerdings mit zunehmender Komplexität und Dynamik der Anforderungen angesichts neuer Technologien und Bedürfnisse ins Stocken geraten ist. Ein aktuelles Architekturmodell setzt dagegen sehr viel stärker auf flexible Integration. Dabei muss gleichzeitig schnelle Innovation unterstützt und die Lauffähigkeit mit Altsystemen gewährleistet werden.

Aus Sicht von PTC ist der Code of PLM Openness (CPO) eine wichtige Weichenstellung, um genau diese Bedingungen langfristig sicherzustellen und im Verbund von Softwareherstellern, Kunden und Systemintegratoren kontinuierlich weiterzuentwickeln.

Als Dietmar Trippner 2011 den CPO vorschlug, hat PTC ohne Zögern mitgemacht und sich aktiv in die Ausarbeitung eingebracht. Eine intensive Debatte führte schnell zur gemeinsamen Einsicht, dass es sich bei dem schnellen Innovationstempo rund um PLM weniger um technische Spezifikationen handeln kann als um eine businessorientierte Beschreibung von Offenheit. Das schlug sich in den Themenfeldern Interoperabilität, Infrastruktur, Erweiterbarkeit, Schnittstellen, Standards, Architektur und Partnerschaft nieder, die jeweils mit wenigen Worten allgemein verständlich definiert wurden, wobei in einer konstruktiven Diskussion zwischen Softwareherstellern, Kunden und Systemintegratoren um jedes einzelne Wort gerungen wurde.

Ein erstes Ergebnis war, dass die Wettbewerber zu einer besser verständlichen Sprache fanden und so im Sinne der Kunden die Qualität der Implementierungen steigern konnten. Weitere Verbesserungen treten nun schrittweise auf Basis des CPO ein. In diesem Jahr startete ein Programm der Zertifizierung, das bei PTC gerade angelaufen ist. Anfängliche Befürchtungen hinsichtlich eines möglichen Verwaltungs-Overheads konnten dabei schnell überwunden werden.

Ein nächster wichtiger Schritt steht für 2018 an: Der Code of PLM Openness wird in eine DIN-Spezifikation überführt. Damit wird noch einmal deutlicher, welche strategische Bedeutung er für alle beteiligten Parteien hat, für die Fertigungsindustrie genauso wie für Systemintegratoren und Softwarehersteller.

Alle Teilnehmer müssen sich dabei bewusst sein, dass die fortschreitenden Innovationen im Bereich der Unternehmenssoftware mit Standardisierungsbestrebungen und dem Ziel der Optimierung hinsichtlich Systemkonsolidierung und Kosten im Dauerclinch liegen. Dieses natürliche Spannungsverhältnis wird sich nie auflösen, und jeder Marktteilnehmer muss für sich entscheiden, an welcher Stelle er sich hier einordnet. Dem Wert der Offenheit steht damit ein Preis gegenüber, der aus dem Einhalten bestimmter Konzepte besteht und der aktiven Teilnahme an der Gestaltung künftiger Konzepte. Denn die Standards von heute sind nicht unbedingt jene, die morgen noch Gültigkeit haben. Das spiegelt sich auch in Aufwänden und Preisen wider, in denen sich später Low-Cost- von Enterprise-Lösungen unterscheiden werden.

