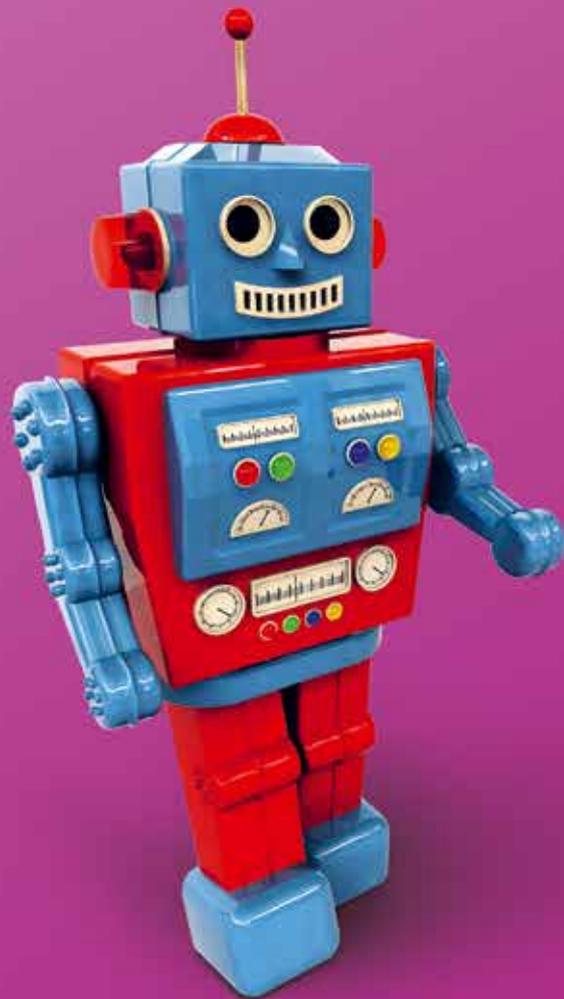


ADNOVUM

# NOTITIA

BEMERKENSWERTES VON UND ÜBER ADNOVUM

FRÜHLING 2017, HEFT NR. 30



DIGITAL EVOLUTION



---

## Inhalt

### **WILLKOMMEN IN DER SOFTWARE-INDUSTRIE 4.0**

Die Digitalisierungswelle erfasst  
auch die Software-Entwicklung 3

### **MIX AND MATCH: DIE NEUEN LÖSUNGSTEAMS**

Entwicklung und Betrieb vereinen ihre Stärken 8

### **DIE DIGITALISIERUNG DES SOFTWARE ENGINEERING**

Automatisierung beschleunigt die Entwicklung  
von Individual-Software 13

### **DIGITALE TRANSFORMATION DURCH BUSINESS ENGINEERING**

Ein Blick auf die Ursachen des Wandels  
und auf Methoden, darauf zu reagieren 18

---

Liebe Leserin, lieber Leser

Maschinen lenken Autos, verwalten Vermögen und führen gar Operationen durch. Möglich macht dies die immer engere Verzahnung von Produktion und Kommunikationstechnik. Das Schlagwort dazu lautet Industrie 4.0 und bezeichnet die vierte industrielle Revolution nach der Dampfmaschine, der Elektrizität und dem Computer. Dank intelligenter und digital vernetzter Systeme soll sich die Produktion selbst organisieren, und zwar über die gesamte Wertschöpfungskette hinweg.

An der Quelle dieser Entwicklung steht die Software-Industrie. Denn: ohne Software keine digitale Automatisierung. Doch wie steht es um die Branche selbst? Sie hat den unschlagbaren Vorteil, dass sie die Herausforderungen mit ihren eigenen Mitteln angehen kann. Nun muss sie ihr Wissen auf die eigenen Prozesse anwenden und diese automatisieren. Ziel ist die vollständige Durchgängigkeit – vom Design über die Entwicklung und das Testing bis zur Produktion. Und wieder zurück.

Eine Schlüsselrolle spielt dabei der Code. Software, die sich immer wieder neu bauen lässt? Alltag. Ein Server auf Knopfdruck? Ebenso. Mit ganzen Systemen «in the Code» eröffnen sich unzählige neue Möglichkeiten. Software wird nicht mehr spezifiziert, entwickelt, verpackt, geliefert, installiert und schliesslich betrieben, sondern in Co-Creation unmittelbar von Kunden, Entwicklern und Betreibern gemeinsam gestaltet und laufend verbessert.

Bei den IT-Anbietern erfordert dies neben der digitalen auch eine personelle Vernetzung. Denn Durchgängigkeit lässt sich nur erreichen, wenn alle beteiligten Teams Hand in Hand arbeiten.

Wir sprechen hier vom sogenannten DevOps-Ansatz. Angesichts der früher strikten Trennung zwischen Entwicklung und Betrieb kommt dieser einem Kulturwandel gleich. Seine Vorteile sprechen für sich: höhere Flexibilität, geringere Time-to-Market.

AdNovum setzt schon länger auf DevOps. Auf diese Weise ermöglicht sie ihren Kunden, Lösungen früh live zu bringen und Schritt für Schritt zu verbessern. Diese stufenweise Optimierung führt zu immer intelligenteren Systemen. Sind diese dereinst vollständig mit all ihren Schnittstellen vernetzt, sind die selbst organisierte Produktion und die Industrie 4.0 alles andere als eine Utopie. Lesen müssen Sie allerdings noch selbst. Dabei wünschen wir Ihnen viel Spass und gute Unterhaltung.

Chris Tanner

CEO AdNovum Informatik AG

# WILLKOMMEN IN DER SOFTWARE-INDUSTRIE 4.0

Die digitale Transformation ist neben dem Internet of Things (IoT) und der Industrie 4.0 eines der grossen Trendthemen in der IT. Sie beeinflusst das Geschäft unserer Kunden ganz entscheidend. Doch nicht nur ihres.

*Von Tom Sprenger und Christian Widmer*

Als unabhängiger Anbieter von Software-Lösungen haben wir über die letzten Jahre bedeutende Veränderungen auf dem Markt generell beobachtet, aber auch spezifisch bei unseren Kunden. Der entscheidende Treiber dahinter ist die Welle der digitalen Transformation, die als globaler Trend alle Branchen erfasst. Um in ihren Märkten wettbewerbsfähig zu bleiben, müssen Firmen Geschäftsmodelle komplett neu denken und in früher nicht vorstellbarer Geschwindigkeit neue Wege gehen. So hat sich zum Beispiel die Erwartungshaltung von Endkunden, wie sie mit Unternehmen kommunizieren können, radikal verändert. Endkunden erwarten heute einen mobilen Zugriff auf Angebote und einen aktiven Austausch. Sie sehen sich als Community auf Augenhöhe mit dem Unternehmen und wollen zunehmend involviert werden.

## ENDKUNDEN WOLLEN ZUNEHMEND INVOLVIERT WERDEN.

Die Geschwindigkeit dieser Entwicklungen stellt unsere Kunden bei der Bereitstellung der dazu notwendigen IT-Lösungen vor immer neue Herausforderungen. Ansätze, wie sie unter den Begriffen «Bimodale IT» oder «Multi-Speed IT» bekannt sind, stossen dabei auf fruchtbaren Boden. Die Zeiten, in denen die IT eine reine Kostenposition darstellte, liegen definitiv hinter uns. Moderne Technologien (zum Beispiel Mobile und Cloud) und Ansätze (zum Beispiel DevOps und Continuous Deployment) in der IT zählen heute unwidersprochen zu den wichtigsten Faktoren, die es erlauben, schnell auf dynamische Kundenbedürfnisse und neue Markttrends zu reagieren.

### Agile Zusammenarbeit und Planungssicherheit

Als Folge dieser Entwicklungen hat sich auch die Zusammenarbeit mit unseren Kunden drastisch verändert. Zum einen hat

das Bedürfnis nach agiler Zusammenarbeit sprunghaft zugenommen. Früher beauftragte uns der Kunde im Sinne eines klassisch wasserfallorientierten Auftragsverhältnisses, schickte uns eine abgenommene Spezifikation und erwartete auf den vereinbarten Zeitpunkt die fertig entwickelte Lösung. Heute möchten uns unsere Kunden bereits während der Spezifikation der Lösung an Bord holen und später die Entwicklung der Lösung kontinuierlich begleiten, damit sie den Kurs laufend an neue Gegebenheiten anpassen können. Zum anderen ist das Bedürfnis unserer Kunden nach (Planungs-)Sicherheit gestiegen. Mit anderen Worten: Unsere Kunden wünschen sich einen Software-

## UM IN IHREN MÄRKTEN WETTBEWERBSFÄHIG ZU BLEIBEN, MÜSSEN FIRMEN GESCHÄFTSMODELLE KOMPLETT NEU DENKEN UND IN FRÜHER NICHT VORSTELLBARER GESCHWINDIGKEIT NEUE WEGE GEHEN.

Delivery-Prozess, der die Agilität, die es für die Fertigung individueller Lösungen braucht, mit den Kerneigenschaften einer hoch industrialisierten Fertigung wie Planungssicherheit, reproduzierbare Qualität, tiefe Stückkosten pro Release und Geschwindigkeit vereint.

### Durchgängige Digitalisierung als Differenzierungsmerkmal

Genau das haben wir uns bei AdNovum zur Aufgabe gemacht: Wir wollen unseren Kunden die Möglichkeit bieten, hoch agil und doch sicher neue Lösungen auf den Markt zu bringen. Dies erfordert eine konsequente Digitalisierung

der Software-Fertigung von der Kundenanforderung bis zur Lieferung. Die durchgängige Digitalisierung ist für uns als Schweizer Software-Haus längerfristig entscheidend, wollen wir die wachsenden Ansprüche des Markts erfüllen. Sie ist für uns eine Chance, uns am Markt zu differenzieren und uns fit zu machen für den zunehmend internationalen Wettbewerb.

### Schritt in digitales Software-Delivery-Modell

Als Voraussetzung für ein erfolgreiches, vernetztes Arbeiten in einem verteilten Ökosystem mit Kunden und Partnern müssen zwei Schlüsselemente gegeben sein: Zum einen benötigen alle beteiligten Parteien ortsunabhängig Zugriff auf dieselbe Informationsbasis. Zum anderen muss sichergestellt werden, dass sowohl Kunden als auch Partner Zugriff auf die wichtigen Prozesse von gemeinsamen Projekten haben und sich aktiv einbringen können. Dies ist eine direkte Analogie zu Digitalisierungsinitiativen, beispielsweise in der Finanzindustrie, in der Endkunden und Beratungspartner heute über den digitalen Kanal Zugriff auf Geschäftsprozesse der Bank erhalten, um im direkten Dialog mit der Bank Geschäfte abzuwickeln.

### Entscheidend ist das digitale Informationsmodell

Ortsunabhängige Verfügbarkeit und gemeinsamer Zugriff auf Prozesse bedingen den konsequenten Wechsel in ein digitales Software-Delivery-Modell. Ein solches Modell besteht im Wesentlichen aus zwei Hauptelementen: einem digitalen Software-Delivery-Prozess und einem digitalen Informationsmodell. Ersterer definiert die Abläufe der wichtigen Aufgaben und Arbeitsschritte von der Kundenanfrage bis zur Lieferung

**BIS HEUTE WERDEN  
VEREINZELT DOKUMENTE  
ODER ANDERE ARTEFAKTE  
PHYSISCH AUSGETAUSCHT.**

der Lösung. Dieser Prozess kann je nach Kundenprojekt verschiedene Ausprägungen haben. Die Varianz ist in diesem Fall unkritisch. Wichtig ist, dass der Prozess klar definiert und digital abgebildet ist sowie im Projekt gelebt wird. Entscheidend für den Erfolg eines digitalen Software-Delivery-Modells ist das digitale Informationsmodell, das heisst, dass alle Arbeitsergebnisse digital und strukturiert abgelegt werden. Sämtliche Prozessartefakte und Lieferobjekte wie Anforderungen, Spezifikationen, GUI-Designs und Tasks werden digital erstellt und bewirtschaftet. Dies klingt in der heutigen Zeit fast schon trivial. Die Praxis zeigt aber, dass das Modell noch lange nicht durchgängig digital umgesetzt ist. So werden bis heute vereinzelt noch Dokumente (zum Beispiel Spezifikationen) oder andere Artefakte physisch ausgetauscht.



Das digitale Software-Delivery-Modell.

### Wichtige Informationen herausziehen

Doch selbst wenn alle Artefakte digital abgelegt sind, heisst das noch lange nicht, dass das digitale Informationsmodell durchgängig umgesetzt ist. Digitale Transformationen scheitern oft daran, dass trotz digitaler Datenflut nicht die wirklich wichtigen Informationen extrahiert werden können. Ein zentraler Teil jeder Digitalisierungsstrategie ist, aus den nach der Transformation verfügbaren Daten Wert zu schöpfen, um im bestehenden Geschäft besser oder schneller zu werden und neue Geschäftsmodelle umzusetzen. Konkret interessiert uns zum Beispiel, wie viele Iterationen wir im Testing durchlaufen mussten, bis das verlangte Qualitätsniveau erreicht war. Dies zeigt auf, wo wir ansetzen müssen, um Verbesserungen zu erreichen. Oder wir möchten herausfinden, ob irgendwo im Code Services mit Wiederverwendungspotenzial implementiert wurden, zum Beispiel Services zum Versenden von Mails. Durch die konsequente Wiederverwendung solcher Services können wir die Time-to-Market verkürzen.

**KUNDEN, PARTNER  
UND ADNOVUM ARBEITEN  
ALS GLEICHBERECHTIGTE AKTEURE  
AUF EINER VIRTUELLEN  
KOLLABORATIONSPLATTFORM.**

### Kunden und Partner: mittendrin statt nur dabei

Die Digitalisierung des Prozess- und Informationsmodells ist Voraussetzung für eine standort- und organisationsübergreifende Zusammenarbeit im Projekt-Ökosystem. Alle für ein Projekt relevanten Informationen sind zu jedem Zeitpunkt für jede berechnigte Person von jedem Ort aus zugänglich. Damit ist zum Beispiel die physische Präsenz bestimmter Personen für eine



Tom Sprenger: Steuert als CTO die digitale Strategie.



Der digitale Business-Prozess.

erfolgreiche Auftragsabwicklung nicht mehr zwingend. Kunden, Partner und AdNovum arbeiten als gleichberechtigte Akteure im Sinne eines projektbezogenen und hoch vernetzten Ökosystems gemeinsam auf einer virtuellen Kollaborationsplattform, die auf dem digitalen Prozess- und Informationsmodell aufsetzt.

Eine solche Kollaborationsplattform beinhaltet sämtliche Funktionen, die für eine digitale Software Delivery benötigt werden, und bietet damit weit mehr als die üblichen Kollaborationsdienste wie Mail, Chat, Task Management und Wikis. Sie ermöglicht damit komplett neue Arbeitsformen, so zum Beispiel die gemeinsame Entwicklung neuer Lösungen, auch bekannt als Co-Creation. Kunden erwerben nicht mehr eine fertige Lösung, die vorgängig definierte Bedürfnisse abdeckt, sondern die Möglichkeit, einen aktiven Part bei der Entwicklung der Lösung zu spielen. Dabei profitieren sie vom Know-how des Entwicklungsunternehmens aus ähnlichen Projekten, insbesondere von massgeschneiderter Projektmethodik und Best Practices.

### DAS KLASSISCHE AUFTRAGGEBER-AUFTRAGNEHMER- VERHÄLTNIS VERSCHWINDET.

#### Co-Creation schafft Transparenz und baut Grenzen ab

Die Zusammenarbeit in digitalen Umgebungen bringt Veränderungen mit sich und erfordert nicht zuletzt auch Anpassungen an der Prozesslandschaft. Das klassische Auftraggeber-Auftrag-

nehmer-Verhältnis verschwindet dabei zu Gunsten deutlich transparenterer und damit am Ende produktiverer Arbeitsformen wie der Co-Creation. Durch den Einsatz moderner Identity- und Access-Management-Mechanismen, wie sie zum Beispiel die NEVIS Security Suite bietet, verschwindet dabei die explizite Unterscheidung von intern und extern bezüglich der partizipierenden Organisationen zunehmend. So können im Rahmen eines Projekts alle involvierten Parteien über Organisationsgrenzen hinweg gleichberechtigt und medienbruchfrei zusammenarbeiten.

**BEI CO-CREATION IST  
DAS FEEDBACK ZU  
ARBEITSSCHRITTEN  
UNMITTELBAR.**

Die praktische Erfahrung mit Kunden zeigt, dass Co-Creation-Ansätze gerade in Phasen mit hoher Unsicherheit die Geschwindigkeit deutlich erhöhen. So zum Beispiel zu Beginn eines Projekts, wenn die Anforderungen noch nicht restlos geklärt sind oder erst eine vage Idee der gesuchten Lösung besteht. Bei einem kollaborativen modellgestützten Ansatz kann bereits in dieser Phase das Wissen aller Teilnehmer genutzt werden und direkt auf das Resultat wirken. Aber auch während der Implementationsphase können in der Kollaborationsumgebung erste Resultate (zum Beispiel GUI-Flows) früh und niederschwellig gezeigt werden. Das Feedback zu Arbeitsschritten ist unmittelbar. Dadurch werden Fehlentwicklungen früh erkannt und korrigiert sowie Iterationszyklen zur Weiterentwicklung von Ideen und Lösungen dramatisch verkürzt. In der Konsequenz verkürzt sich als weiterer Effekt die Time-to-Market von der Idee über die Umsetzung bis zur produktiven Einführung, was wiederum die Entwicklungskosten senkt.

**DEN GRÖSSTEN HEBEL  
BIETEN MANUELL ERSTELLTE  
ARTEFAKTE.**

**Digitale Information erzeugt Mehrwert**

Werden sämtliche im Software-Delivery-Prozess erzeugten Artefakte im Modell abgelegt, können sie ohne zusätzlichen Aufwand laufend analysiert werden, was eine kontinuierliche Verbesserung erlaubt. Den grössten Hebel bezüglich Optimierung bieten manuell erstellte Artefakte wie Spezifikationen, Projektstrukturen, Source Code oder ganze Infrastrukturlandschaften. Dabei kann man zwischen Artefakten unterscheiden, die zwingend eine kreative Leistung erfordern, und solchen, die



*Christian Widmer: Implementiert die Strategie in Asien.*

redundant erstellt oder aus vorhandenen Artefakten abgeleitet werden und somit weitgehend in reiner Fleissarbeit umgesetzt werden können.

Um brachliegendes Automatisierungspotenzial auszuloten, können wir in einem ersten Schritt die Informationen aus dem digitalen Software-Delivery-Modell nutzen. Wir ermitteln hierfür im Modell erst Aufgaben ohne kreativen Anteil und erzeugen danach aus den semantisch annotierten Artefakten Folgeartefakte. So lassen sich zum Beispiel aus einer Businessspezifikation architektonische Basiselemente extrahieren und daraus dann sowohl das projektspezifische Gerüst der Lösungsarchitektur als auch die Grundstruktur des Software-Projekts sowie dessen Systemdeployment automatisiert erstellen. An dieser Stelle sei explizit erwähnt, dass es dabei nicht um das Generieren von Businesslogik geht – diese bedarf in den meisten Fällen kreativer Kapazität und entzieht sich damit dem klassischen Generierungsansatz.

### Fertigungstiefe reduzieren

Das Erstellen von Artefakten, die eine kreative Leistung erfordern, ist an Brain Power und damit an bestimmte Personen gebunden. Somit skaliert ihre Erstellung nur bedingt und sie sind entsprechend wertvoll. Heute werden solche Artefakte oft noch auf Papier oder in proprietären Formaten erstellt und abgelegt. In einem digitalen Software-Delivery-Modell sind sie zu jedem Zeitpunkt in digital aufgeschlüsselter Form und semantisch verknüpft verfügbar. Dadurch eröffnen sich neue Möglichkeiten.

## ARTEFAKTE, DIE EINE KREATIVE LEISTUNG ERFORDERN, SIND AN BRAIN POWER GEBUNDEN.

So können zum Beispiel durch entsprechendes Tooling ähnliche Artefakte inklusive ihres Kontexts sichtbar und verfügbar gemacht werden. Beispielsweise ist es möglich, beim Bau einer neuen Lösung vorgefertigte Bausteine aus einem bestehenden Service Repository zusammenzufügen. Durch die Reduktion der Fertigungstiefe lässt sich das mehrfache Erstellen von gleichen oder ähnlichen Artefakten weitgehend vermeiden. Dies setzt wertvolle kreative Kapazität frei, die an anderer Stelle höheren Mehrwert bringt.

### Digitalisierung setzt Ressourcen frei

Die durchgängige Digitalisierung des Software-Delivery-Modells ermöglicht neue effizientere Formen der Zusammenarbeit sowie eine zusätzliche Automatisierung sowohl bei redundanten als auch bei kreativen Tasks. Sie setzt dadurch Ressourcen frei und erhöht gleichzeitig die Agilität in der Projektarbeit. Es wird damit deutlich einfacher, im Rahmen eines Projekts in

enger Zusammenarbeit mit einem Kunden innovative Ansätze auszutesten. Auf diese Weise lässt sich stetige Innovation ohne grosses Risiko vorantreiben.

## DAS DIGITALE SOFTWARE-DELIVERY-MODELL LEGT DIE BASIS FÜR INNOVATION IN FORM KOMPLETT NEUER ANGEBOTE.

Nike macht es uns heute bereits vor. Kunden können über das Internet persönlich gestaltete Nike-Turnschuhe (NIKEiD) bestellen. Nike verknüpft dabei einen hoch automatisierten Herstellungsprozess mit der Option, das Resultat individuell zu gestalten. In der Fachsprache nennt man das auch individuelle Massenproduktion und Digital Fabrication. Der Schritt in ein digitales Software-Delivery-Modell schafft die Grundlage für analoge Möglichkeiten in der Software-Industrie, so dass sowohl wir als auch unsere Kunden neue Chancen im Markt schneller gezielt wahrnehmen können. Er legt aber auch die Basis für Innovation in Form komplett neuer Angebote – willkommen bei Software Engineering 4.0! ■

### Tom Sprenger

*Tom Sprenger, Dr. sc. techn. ETH, kam 2000 als Software Engineer zu AdNovum. Ab 2002 leitete er die Schwesterfirma AdNovum Software Inc. in San Mateo, CA. Zurück in der Schweiz, wurde er 2007 zum Chief Information Officer (CIO) und zum Mitglied der Geschäftsleitung ernannt und baute den strategischen Geschäftsbereich IT Consulting auf. Seit 2013 ist er als Chief Technology Officer (CTO) für die Technologiestrategie – mit dem zentralen Thema Digital Evolution – und für die Business-Unit-Produkte verantwortlich. Wenn er mal nicht arbeitet, genießt er seine Familie und im Winter sonnige Tage auf den Skipisten.*

### Christian Widmer

*Christian Widmer, Dipl. Informatik-Ing. ETH, ist seit 2002 bei AdNovum tätig. Als technischer Projektleiter hat er eine Reihe von technisch komplexen Enterprise-Projekten betreut. 2010 zog er nach Singapur, um beim Aufbau der Schwesterfirma AdNovum Singapore Pte. Lte. mitzuhelfen. Als Regional CTO ist Christian Widmer seit 2013 massgeblich für die Definition und die Realisierung von AdNovums Strategie im asiatischen Markt zuständig. Abseits der Arbeit genießt er als leidenschaftlicher Vater das Familienleben.*

# MIX AND MATCH – DIE NEUEN LÖSUNGSTEAMS

Die fortschreitende Digitalisierung stellt die Prozesse im Software Engineering auf den Kopf. Denn Kunden bestellen heute nicht mehr fertige Software, sondern eine kontinuierliche Digitalisierung ihrer Wertschöpfungskette. Gut, vereint AdNovum Entwickler und Betreiber unter einem Dach, damit sie ihre Lösungsteams flexibel kombinieren kann. Interview mit Christian Siffert und Alexander Nolting.

Wie haben sich die Bedürfnisse unserer Kunden verändert, wenn wir ihre Bestellungen betrachten?

CS: Ganz wesentlich. Früher bestellte der Kunde Software auf CD oder per Download und betrieb sie selbst. Heute bestellt er einen ganzen Service. Die Wertschöpfungskette hat sich verbreitert, die Kadenz verkürzt. Lieferten wir früher zweimal im Jahr einen Release, so geschieht dies heute im Wochentakt.

AN: Die Wertschöpfungskette beginnt auch früher. Die Digitalisierung fängt vor dem ersten Zusammentreffen mit dem Kunden an. Dabei werden bestimmte digitale Prozesse vorausgesetzt, damit Kunde und AdNovum miteinander über Plattformen kommunizieren können.

**BISHER WAREN WIR EINE  
MANUFAKTUR. NUN DESIGNEN WIR  
KOMPLETTE PRODUKTIONSLINIEN  
ZUSAMMEN MIT DEM  
KUNDEN.**

Bedeutet das, dass unsere Kunden heute ihre Anforderungen schon in digitaler Form liefern, zum Beispiel über Kollaborationstools?

AN: Unsere Kunden liefern die Anforderungen immer digital, aber nicht zwingend über Kollaborationstools. Es gibt nach wie vor das klassische Word-Dokument. Das ist je nachdem sogar effizienter, da Kollaboration noch nicht über Unternehmensgrenzen hinweg vereinheitlicht ist. Denn jedes Unternehmen nutzt Kollaborationstools anders.

CS: Früher bildete ein umfangreiches Dokument mit Anforderungen den Ausgangspunkt für unsere Arbeit. Zwar tauschten wir uns laufend mit dem Kunden aus, um Anforderungen

anzupassen und Detailfragen zu klären. Bis er aber ein Resultat in Form lauffähiger Software erhielt, verstrich eine längere Phase. Heute läuft alles viel schneller. Bereits während der Umsetzung werden einzelne Features dem Kunden übergeben. Die Teilprozesse sind enger verzahnt, wir produzieren «just in time», wie man das in der Industrie nennt. Viele Veränderungen, die die Industrie in der Nachkriegszeit erlebte hat, finden heute auch in der IT statt. Während wir bisher eher eine Manufaktur waren, die hier und dort einen Release auslieferte, designen wir nun komplette Produktionslinien zusammen mit dem Kunden, die kontinuierlich Features in kleinen Batchgrößen liefern.

Es gibt also nicht mehr einen Anfang und ein Ende eines Projekts?

CS: Ein Projekt hat per Definition einen Anfang und ein Ende. Man bildet ein Team und löst es wieder auf. Heute denken wir nicht mehr in Projekten, sondern in Produkten oder Lösungen. Die haben kein Ende.

Was verkaufen wir denn genau?

CS: Unsere Kunden bestellen kein Software-Projekt mehr, sondern die Digitalisierung ihrer Wertschöpfungskette. Sie wollen eine kontinuierliche Lösung.

AN: Die Lösung ist zudem häufig nicht mehr auf das anfänglich gesteckte Ziel fokussiert. Denn dieses kann morgen schon anders sein. Die Kunden erwarten, dass wir Veränderungen rasch aufnehmen. Wir müssen uns in ihre Arbeits- und Gedankenwelt hineinversetzen, um herauszufinden, wo sie hinwollen und ob die Lösung eventuell weitere Aspekte ihres Geschäftsmodells berührt.

CS: Unsere Kunden kennen die kurzen Zyklen bei den Consumer Devices: sehen, bestellen, erhalten. Alles innerhalb einer Woche. Deshalb fragen sie sich: Warum soll das bei einem digitalisierten Unternehmensprozess nicht gehen?



Alexander Nolting und Christian Siffert: Bringen Betrieb und Entwicklung auf einen Nenner.

Das Tempo hat enorm zugenommen, die Software wird laufend weiterentwickelt. Was heisst das für die Zusammenarbeit von Entwicklung und Betrieb?

AN: Entwickler müssen auf einem produktionsnahen System entwickeln können, das auf die Anforderungen des Kunden zugeschnitten ist, keinen Unterschied zwischen Entwicklung und Betrieb macht, getestet ist und Support- und Monitoring-Prozesse bereithält. Denn letztlich muss die Software nicht nur auf dem Entwickler-Laptop, sondern auch im Betrieb laufen.

CS: Es gibt den Spruch «Der Entwickler wirft die Software über den Zaun». Früher war dies eine manuelle und explizite Übergabe von Software und Installationsdokumenten. Ein Betriebsmitarbeitender musste die Software auf dem Produktionssystem installieren und schauen, dass alles richtig lief. Das war zwar aufwändig, da es aber nur alle paar Monate geschah, war es akzeptabel. Wenn das dauernd gemacht werden muss, tut es richtig weh. Das Ausrollen auf die Systeme muss deshalb automatisiert erfolgen. Dies bewirkt eine Verschiebung in der Wertschöpfungskette. Früher waren die beiden Arbeitsschritte sequentiell verkettet, heute laufen sie parallel ab. Der Betrieb stellt einen Service bereit, den der Entwickler nutzt.

Mit anderen Worten: Betrieb und Entwicklung fahren heute Tandem und übergeben nicht mehr wie bei einer Stafette den Stab?

CS: Genau. Kürzlich benutzte ich für einen Blog Post ein Bild: Für einen Release belädt man einen Lastwagen und geht auf

die Strasse. Dabei hat man einen Pace Car vor sich, dem man zum Deployen hinterherfahren muss. Das funktioniert heute nicht mehr, da es zu langsam ist. Der Betrieb muss eine sauber geplante Autobahn mit Leitplanken bereitstellen. Doch nachher muss das Lösungsteam autonom sein. Das ist eine grosse Umstellung für beide Seiten.

AN: Vieles hängt vom Know-how des Einzelnen ab. Ein Entwickler beispielsweise, der gute Algorithmen schreibt, sich aber nicht für die Infrastruktur interessiert, in der die Lösung läuft, wird scheitern. Es ist deshalb notwendig, über den Tellerrand hinauszublicken.

CS: Ja, das ist heute zwingend. Das Team muss den Kundenwert als Ganzes im Blick behalten, inklusive Betrieb. Dies bedingt T-Profile und gemischte Teams und, wie gesagt, eine stärkere Abstützung auf unterliegende Self-Service-Plattformen.

### DER BETRIEB MUSS EINE SAUBER PLANIERTE AUTOBAHN BEREITSTELLEN.

Was genau bietet der Betrieb?

AN: Der Betrieb sorgt für die Stabilität von Prozess und Lösung. Er ist nicht zwingend für den Prozess verantwortlich, weiss aber, was mit den physischen und personellen Ressourcen betreibbar ist.

CS: Der Betrieb kennt sich mit der Implementierung von



Prozessen aus. Dieses Wissen muss in die Entwicklungsprojekte einfließen. Statt wie bisher von Entwicklungsteams würde ich deshalb heute von Lösungs- oder Produktteams sprechen. Sie haben sowohl eine Entwicklungs- als auch eine Betriebsverantwortung.

AN: Da Entwicklungs- und Betriebsteams oft separat agieren und beide auf ihre Aufgaben fokussiert sind, haben sich das interdisziplinäre Arbeiten und Denken in der IT allerdings noch nicht gänzlich durchgesetzt.

CS: Dies hat natürlich massive organisatorische Konsequenzen. Bisher unterstanden Entwicklungsteams einem Entwicklungschef und Betriebsteams einem Betriebschef. Nach der neuen Methodik werden plötzlich Lösungsteams gebildet, die die Verantwortung für beides tragen. Der Slogan dazu lautet: «You build it – you run it.» Schauen wir es von Kundenseite an: Ein Kunde tritt mit einer bestimmten Anforderung an uns heran. Was er von uns will, ist nicht Software, sondern Leistung oder einen «Value». Ob Autoleasing oder Zementtransport: Er will sein Business organisiert haben. Wer und was AdNovum-intern von der Business-Analyse bis zum Betrieb noch alles involviert ist, interessiert ihn nicht.

## WIR HABEN EINEN REIFEGRAD IN IT OPERATIONS ERREICHT, DER VOR FÜNF JAHREN NOCH UNDENKBAR SCHIEN.

**Das bedeutet, die Integration von Entwicklung und Betrieb unter einem Dach ist fast zwingend?**

AN: Für ein Unternehmen, das Lösungen von der Stange anbietet, ist die Unterscheidung zwischen Betrieb und Entwicklung nicht matchentscheidend. AdNovum jedoch baut massgeschneiderte Lösungen. Wenn wir das komplette Angebot liefern wollen, von der Beratung über die Konzeptentwicklung bis hin zu Betrieb und Optimierung, kommen wir nicht umhin, die betrieblichen Prozesse in die Entwicklung zu integrieren.

CS: Die Integration der beiden Bereiche ist auch eine Effizienzfrage, denn wir sind nicht die Einzigen auf dem Markt. Vielleicht bezeichnen wir uns auch irgendwann nicht mehr als Softwareentwicklungsunternehmen, sondern als Unternehmen für «Informationsautomatik». Software Engineering bleibt zwar ein wichtiger Arbeitsschritt. Gemessen werden wir aber an der reibungslosen Funktion im Betrieb sowie einer kurzen Durchlaufzeit vom Kundenwunsch bis zur Lieferung.

AN: Der Anspruch des Kunden an sich selbst spielt ebenfalls eine Rolle. Bei einem bundesnahen Unternehmen stehen beispielsweise eine lange Lebensdauer, Stabilität sowie niedrige Ausfall- und Fehlerraten im Vordergrund. Firmen, die zwingend dynamisch agieren müssen, legen dagegen unter Umständen die Toleranzschwelle bei der Betriebssicherheit höher, wenn es dadurch schneller geht.

**Wie lässt sich eine solche Flexibilität erreichen?**

AN: Wir haben einen Reifegrad in IT Operations erreicht, der vor fünf Jahren noch undenkbar schien. Tools entstehen mit derart rasanter Geschwindigkeit, dass es nahezu unmöglich ist, ihre Fülle im Auge zu behalten. Noch schwieriger ist die Entscheidung, welches Tool das richtige ist. Hier müssen wir auch wieder standardisieren, sonst verzetteln wir uns in Optionen.

CS: Der Bedarf nach Flexibilität treibt in der IT heute die Standardisierung durch Container voran. In der Schifffahrtsindustrie wurde der Container schon in den 60er-Jahren eingeführt. Kostete die Verschiffung einer Tonne 1956 noch 6 Dollar, waren es nach der Containerisierung gerade noch knapp 20 Cents. Die Flexibilität steigt etwas durch kürzere Durchlaufzeiten, zudem sinken die Kosten enorm.

**Übertragen auf Software heisst das ...?**

CS: Die Standardisierung von Komponenten setzt Mittel frei, die mehr Flexibilität bei gleichzeitig hoher Qualität erlauben. Wenn man beispielsweise die hohen Aufwände für sich wiederholende manuelle Tätigkeiten in Entwicklung, Lieferung, Integration und Inbetriebnahme anschaut, wird das Potenzial für Verbesserungen ersichtlich.



AN: Ich muss verschiedene Fragen beantworten können: Wie binde ich den Kunden an? Hat er bei uns schon einen Fuss in der IT, sodass ich den Container in seine Infrastruktur verschieben kann? Oder muss ich den Container auf anderem Wege bereitstellen? Aus betrieblicher Sicht sind die Container nur ein Schritt. Es ist eine andere Darreichungsform von Software. Es gibt weiterhin Aspekte, über die ich mir Gedanken machen muss. Wir haben Tools, die in die Public Cloud verschieben, doch ist die Public Cloud im Grunde auch nichts anderes als eine grosse Infrastruktur, die von jemandem betrieben wird. Und vielleicht erlaubt der Kunde die Nutzung einer Public Cloud nicht. Dann kann AdNovum als Schnittstelle fungieren und die Kundenlösung in ihrer eigenen Cloud betreiben. Oder wir entwickeln im herkömmlichen Sinne und betreiben einen ganzen Betriebs-Stack in einer virtuellen Maschine. Das wäre auch eine Form von Übergabe. Containerisierung löst solche Fragen nicht, sondern bietet zunächst einen neuen Weg an. Für den Betrieb bieten sich dadurch allerdings andere, ganz neue Möglichkeiten, da einige bisher vorhandene Herausforderungen bei richtigem Einsatz wesentlich besser gehandhabt werden können.

## HEUTE IST DIE SERVER-PFLEGE NUR NOCH AUTOMATISIERT MÖGLICH.

### Noch mal zurück zur Bedeutung für die Zusammenarbeit von Entwicklung und Betrieb.

AN: Für den Entwickler ändert sich durch das Containerparadigma einiges. Er beschreibt den Betrieb formal und wird damit sozusagen zum Betreiber seiner Software. Der Betreiber dagegen entwickelt sich zunehmend zu einem (Performance-)Analysten, der seine Infrastruktur mit Blick auf Stabilität und Performance optimiert.

CS: Wie gesagt, wird Infrastruktur heute vermehrt als Service angeboten, die einzelne Bestellung also automatisiert und nicht mehr einzeln durch einen Betriebsmitarbeitenden behandelt. Dies macht Lieferanten und Nutzer autonomer. Zudem setzt es –

hoffentlich – Zeit frei, sodass Betriebsmitarbeitende und Entwickler tatsächlich zusammenarbeiten, das heisst gemeinsam Massnahmen zur Verbesserung von Verfügbarkeit, Ausfallsicherheit, Monitoring und konkrete Incident-Massnahmen erarbeiten können, anstatt sich nur sporadisch in Meetings zu treffen.

### Früher hiess es, eine Trennung zwischen Entwicklung und Betrieb sei effizienter und sicherer. Gilt das nicht mehr?

CS: Diese Aussage bezog sich auf den Grundsatz der «Separation of Concerns», der oft der IT Infrastructure Library (ITIL) zugeschrieben wird. Doch faktisch steht das in der nirgends drin. Angesichts der heutigen Rahmenbedingungen und Bedürfnisse erachte ich diese Trennung nicht mehr als den effizientesten Weg.

AN: ITIL unterstützt durch sogenannte Blaupausen, wie man etwas organisieren muss, um gute Resultate zu erhalten. Die Trennung muss jedoch nicht so strikt sein.

CS: Hier sind Veränderungen notwendig. Ein interdisziplinäres Team, das spezifische Fähigkeiten in sich vereint, bringt die Lösung vorwärts. Ein Entwickler weiss, wie man codiert, ein Betreiber weiss, wie die Systeme funktionieren. Als Beispiel: Früher hatte man in der Produktion vielleicht bis zu einem Dutzend Server. Heute sind es 100. Früher wurden die einzelnen Server wie Haustiere gehegt, heute ist die Pflege nur noch automatisiert möglich. Es braucht dafür einen Austausch von Fähigkeiten zwischen Entwicklung und Betrieb, und zwar in beide Richtungen. Der Entwickler hat beispielsweise schon immer versioniert. Das passiert heute auch mit Infrastruktur. Technisch sind wir verstärkt unter Druck, alle verfügbaren Ressourcen wie Firewall und Storage zu versionieren, automatisch zu reproduzieren und damit Fehler zu vermeiden.

### Was bedeutet das für die Arbeit von Entwicklern und Betreibern?

AN: Das Handwerksset wird sich ändern. Ich persönlich setze zum Beispiel im Bereich Systemintegration voraus, dass die Mitarbeitenden entwickeln können. Sie müssen verstehen, wie Entwickler arbeiten, um Probleme effizient zu lösen. Umgekehrt kann sich der Entwickler heute nicht mehr nur auf das Coding konzentrieren, sondern muss sich auch Gedanken dazu machen,



wie die Lösung nachher läuft. Am deutlichsten sieht man dies beim Umgang mit Datenbanken. Früher hat der Entwickler die Datenbank und deren Setup beim Datenbankadministrator bestellt. Heute, wo die Datenbank im Container betrieben wird, macht er sie häufig selbst. Er muss also auch ein gewisses DBA-Know-how besitzen.

CS: Der Betrieb sitzt nicht mehr alleine am Ende der Wertschöpfungskette und setzt auf Zuruf manuell eine Datenbank auf. Er arbeitet vermehrt im Projektmodus. Und die grosse Menge an Hardware, von Servern über Firewall und Load Balancer, wird zur automatisierten Bewirtschaftung mit Konfiguration oder Code beschrieben.

AN: Du hast das schön umschrieben mit «Infrastruktur als Code». Wir müssen Infrastruktur so formalisiert beschreiben, dass wir daraus Code ableiten können. Welche formale Sprache wir dafür verwenden, spielt keine Rolle.

## ENTSCHEIDEND IST, WIE WIR MIT FEHLERN UND AUSFÄLLEN UMGEHEN.

### Das klingt anspruchsvoll ...

AN: Entscheidend ist, wie wir mit Fehlern und Ausfällen umgehen. Dass sie passieren, ist klar. Ein Aspekt des Containergedankens sagt: «Wenn ich ausfalle, nimm eine Kopie von mir in Betrieb. Die macht genau da weiter, wo ich ausgefallen bin.» Das heisst, wir leben mit der Gewissheit, dass Fehler passieren, und planen sie ein.

CS: Das ist ein Riesen-Shift. Heute ist der Betrieb darauf optimiert, dass keine Fehler passieren. Wir wissen aber, dass in der IT wie bei der Globalisierung alles mit allem zusammenhängt. Aufgrund der stärkeren Abhängigkeiten treten Ausfälle viel häufiger auf – und unter Umständen wegen eines Services, von dem ich noch nicht einmal wusste, dass er einen Einfluss auf meine Lösung hat. Wir müssen vermehrt dahingehend optimieren, schnell wieder einsatzfähig zu sein. Hier sind Entwicklung und

Betrieb beziehungsweise Lösungsteams gefordert, gemeinsam diese Stabilität zu schaffen.

### Wie macht ihr das bei AdNovum?

AN: Wir investieren in eine Kultur, die akzeptiert, dass Fehler passieren, und Wege sucht, die Software wieder zum Laufen zu bringen. Das erfordert ein radikales Umdenken, denn das klassische Vorgehen in der Software-Entwicklung ist, Komplexität zu reduzieren, um den Prozess in den Griff zu bekommen. Das funktioniert aber heute nicht mehr.

CS: Das Thema hat auch organisatorische Konsequenzen, denn Entwicklung und Betrieb sind heute getrennt. Als Matrixorganisation müssen wir jedoch die Organisation deshalb nicht komplett umkrempeln, sondern können von Fall zu Fall entscheiden, ob es sinnvoll ist, ein gemischtes Lösungsteam einzusetzen. So können wir Erfahrung sammeln und objektiv beurteilen, welche Vor- und Nachteile dies für uns und unsere Kunden hat. ■

---

### Christian Siffert

*Christian Siffert, Dipl. Informatik-Ing. ETH Zürich, verantwortet seit dem Frühling 2016 den Platform-Infrastructure-Engineering-Bereich der AdNovum. Er ist seit 1999 bei AdNovum tätig. Als Entwickler und technischer Projektleiter sammelte er Erfahrung in zahlreichen Projekten. Seit Lean Production in der IT zum Thema geworden ist, lässt er auch gerne Gedanken aus dem Nebenfach Logistik einfließen oder besucht Lean-Kanban-Konferenzen. Zum Ausgleich lüftet er seine Gedanken beim Joggen im Jura-Gebirgszug oder bei Hobbys mit seiner Familie.*

### Alexander Nolting

*Alexander Nolting, Informatiker und Projektleiter, kam 2016 als Verantwortlicher des Bereichs System Integration zu AdNovum. Er besitzt nahezu 20 Jahre Erfahrung als Projektleiter für hoch verfügbare Systeme in der Telekommunikation. Wenn er nicht arbeitet, bewegt er sich mit dem Töff durch die Lande.*

# DIE DIGITALISIERUNG DES SOFTWARE ENGINEERING

Ohne Software geht heute fast gar nichts mehr. Die Anforderungen an Verfügbarkeit, Stabilität und Flexibilität steigen laufend, effizientere Produktionsmethoden sind gefragt.

Die Software-Industrie steht vor dem Schritt, den die meisten Branchen längst hinter sich haben – sie muss ihre Prozesse digitalisieren.

*Von Remo Meier und Stefan Ott*

Schneller, weiter, höher. Wie Sportler werden auch Unternehmen zunehmend auf Höchstleistung getrimmt. Effizienz und Produktivität sollen steigen, die Kosten sinken, Produktänderungen und -neuerungen rasch umgesetzt und eingeführt werden. Hierfür müssen Abläufe und Prozesse auf allen Ebenen optimiert werden, indem die Übergabe von Zwischenprodukten zwischen den Produktionsstufen harmonisiert wird.

Ein Lösungsweg liegt nahe: diese Ziele mit Hilfe von IT zu erreichen. In vielen Branchen ändert sich die Beziehung zur IT fundamental. IT-Beschaffungen sind längst nicht mehr fixe Beschaffungen analog zu Wasser und Strom. Ganz im Gegenteil, die IT entwickelt sich zum integralen Bestandteil eines Unternehmens. Als solcher entwickelt sie sich mit dem Unternehmen zusammen ständig und rasch weiter.

Durch ihre neue Beziehung zur IT orientieren sich Branchen zunehmend an erfolgreichen IT-Konzepten wie Mobile Apps und dem Internet of Things (IoT). Dies eröffnet neue Möglichkeiten. Unternehmen können Produkte zur Plattform ausbauen, indem sie deren Schnittstellen für Endkunden und Drittfirmen verfügbar machen. Dies ermöglicht Innovation von aussen, was zu einem exponentiellen Wachstum und Erfolg führen kann.

Kurz: Es wird viel Software benötigt. Das stellt IT-Unternehmen vor diverse Herausforderungen: Sie müssen die Entwicklung der Software vom Design bis zum Betrieb verbinden, die Produktion verbessern und beschleunigen, Medienbrüche eliminieren und einen raschen Informationsaustausch ermöglichen, um die Anforderungen des Markts zu erfüllen.

## Vom Projekt zur Lösung

Aus dem vermehrten Einsatz und der höheren Prominenz von IT ergeben sich der Wunsch nach und eine neue Form der Zusammenarbeit zwischen dem Anwender/Kunden und seinem IT-Dienstleister. Im Vordergrund steht dabei ein enger und effizienter Kontakt, der von der Analyse und Konzeption bis zum

produktiven Betrieb und zur Weiterentwicklung reicht. Auch der Auftraggeber ist heute oft ein anderer. War vor einigen Jahren noch die interne IT des Kunden der typische Auftraggeber, ist es heute in der Regel der Fachbereich selbst, der beispielsweise eine Applikation bestellt. Der Kunde hat viele Ideen und möchte diese einbringen. Um sie zu evaluieren und zu kommentieren, sind schnelle Lösungen gefordert. Dasselbe gilt für Änderungen. Sie sollen möglichst rasch berücksichtigt werden, damit der IT-Anbieter auch das entwickelt, was der Kunde sich wünscht. In einem solchen Kontext wandelt sich ein Projekt mit begrenzter Laufzeit zu einer Lösung mit der Laufzeit des ganzen Software-Lebenszyklus, also typischerweise Jahren. Software als Service im besten Sinne. Die Folge: Der Anspruch an Integration und Güte der Software steigt.

## INNOVATION VON AUSSEN KANN ZU EINEM EXPONENTIELLEN WACHSTUM UND ERFOLG FÜHREN.

### Von der digitalen Produktionsinsel zum digitalen Förderband

Was ist nun der nächste Schritt? Im Vordergrund steht die Automatisierung der Prozesse. Ein entscheidender Vorteil und mit Potenzial verbunden ist die Tatsache, dass die Software-Industrie ihre Herausforderungen mit eigenen Mitteln angehen kann. Methodiken und Fachwissen sind vorhanden. Nun ist die Branche gefordert, dieses Wissen auf ihre eigenen Prozesse anzuwenden mit dem Ziel, sie neu zu gestalten und zu automatisieren.

Bedeutend dabei ist, dass der Lebenszyklus einer Software weit mehr als Entwicklung umfasst. Es geht darum, Einheitlichkeit und Durchgängigkeit zu schaffen, und zwar vom Design über die Entwicklung und das Testing bis hin zur Produktion und



*Stefan Ott (links) und Remo Meier beschreiben, wie Automatisierung die Entwicklung von Individual-Software beschleunigt.*



wieder zurück. So helfen zum Beispiel die folgenden Massnahmen, die Durchlaufzeit zu verkürzen: Deployments automatisieren, technische und organisatorische Massnahmen für Zero Downtime etablieren und abstimmen, Risk Management betreiben, Service Availability und Performance sicherstellen, Performance tracken und nicht zuletzt Qualität gewährleisten. Enge Zusammenarbeit mit dem Betrieb und präventive Massnahmen sollen dazu beitragen, dass ein System möglichst wartungsfrei wird. Kollaborationsplattformen beispielsweise beseitigen Barrieren zwischen beteiligten Organisationen, erleichtern den Austausch und machen Informationen jederzeit zugänglich.

### Der Code auf Expansionskurs

Dabei müssen wir nicht bei null beginnen. Die Automatisierung von Software-Engineering-Leistungen ist seit Jahren im Gang. Das zeigt ein Blick auf den Code. Dieser umfasst heute weit mehr als bloss den eigentlichen Applikations-Code und Code für das «System». Dokumentation wird als Code mit Markup-Sprachen abgelegt. Grosse Teile der Infrastruktur von Servern, Netzwerken und Storage werden per Code definiert sowie automatisiert erstellt und unterhalten («Infrastructure as Code», DevOps). Auch Teile der Integration und des Testings werden automatisiert und zu Code. Hinzu kommt das vollautomatische Deployment von Software, ohne Downtime und unter Einbezug und vollautomatischer Steuerung von Load Balancern, oder die elastische Skalierung von Systemen wie Servern, Applikationen und Nodes aufgrund des Lastverhaltens, das in Monitoring-Systemen ersichtlich ist (automatisierte Rückkopplung).

## CODE UMFASST HEUTE WEIT MEHR ALS BLOSS DEN EIGENTLICHEN APPLIKATIONS- UND SYSTEM-CODE.

### Mit Code zur Reproduzierbarkeit

Mit ganzen IT-Systemen «in the Code» eröffnet sich eine Vielzahl neuer Möglichkeiten. So wird Code seit Jahren in Versionsverwaltungssystemen abgelegt. Dies ermöglicht unter anderem die Archivierung, die Einsicht in und den Vergleich von Versionen sowie die Rückführung zu früheren Zuständen. Von ganzen Systemen kann eine Kopie erstellt, weiterentwickelt und dann zurück in den Hauptentwicklungszweig integriert werden. Der Entwicklungszweig bleibt dabei voll funktionsfähig und weitere Änderungen lassen sich jederzeit schnell bis in die Produktion ausbreiten.

Für die Durchgängigkeit und die Automatisierung von der Entwicklung bis in die Produktion ist es von Vorteil, alle weiteren Einflüsse zu eliminieren. Prozesse und der Zustand der Systeme sollen in allen Umgebungen möglichst gleich sein. Dies ermög-

licht es, Probleme frühzeitig zu erkennen oder sogar ganz zu vermeiden. Ansätze wie Immutable Infrastructures nehmen sich dieser Konzepte an: So werden Server nicht mehr im klassischen Sinne installiert, sondern aus dem Code fertige Images erstellt, die keine weiteren Änderungen an sich selbst erlauben. Diese Images werden automatisiert an die Server ausgeliefert, installiert und gestartet. Der Prozess ist für die Test- und die produktive Umgebung identisch. Die Konfiguration erhalten die Images aus ihrer Umgebung. Die Datenablage von Business Records bis Auditing und das Logging erfolgen in zentralen Datenbanken. Die statische Natur der Images erlaubt vielfältige Vereinfachungen an der Infrastruktur. Aspekte wie Firewalls, Fernzugriff, Benutzerkontrollen und File-System-Berechtigungen werden in vielen Bereichen irrelevant. Das genau gleiche System lässt sich jederzeit wieder aus dem Code erstellen und starten, genauso wie sich eine Software immer wieder neu bauen lässt. Jeder Schritt im System wird reproduzierbar. Es sind keine manuellen Schritte mehr nötig. All dies steigert die Effizienz, die Qualität und die Verfügbarkeit. Ein System ist stets exakt durch seinen Code definiert. Und eine einzelne Versionsnummer genügt, um den Code zu identifizieren.

### Von Servern zu Events

Wieso hier stoppen? Der Unterhalt einer Server-Landschaft ist mit erheblichem Aufwand verbunden. Ist das System erst einmal im Code definiert und sind weitere Einflüsse eliminiert, lässt sich die Automatisierung einen bedeutenden Schritt weiterbringen. Inwieweit ist ein Server, den man installiert, startet und unterhält, noch notwendig, wenn sich dieser Server jederzeit aus dem Code neu erstellen lässt, in der extremsten Form gar für jede neue Anfrage oder jede Batch-Verarbeitung? Anstatt von Servern spricht man hier deshalb von Events. Die Vorteile des Event-Ansatzes liegen auf der Hand:

- Installation und Starten von Servern entfallen.
- Infrastruktur wird nur beansprucht, wenn sie verwendet wird, und skaliert mit der Verwendung.
- Redundanz und Fehlertoleranz sind automatisch und transparent verfügbar. Events werden auf der Basis aktueller Last, geografischer Verteilung, verfügbarer Hardware und Anforderungen an die Code Performance dynamisch auf Maschinen verteilt.
- Housekeeping von Servern entfällt.
- Es gibt keine Server-Prozesse, die über Monate und Jahre aufrechterhalten werden müssen.

Verschiedene Faktoren beeinflussen eine erfolgreiche Anwendung des Event-Ansatzes. Im Alltag unerlässlich und technisch gut zu realisieren ist beispielsweise ein durchgängiges und in vielen Bereichen automatisiertes Monitoring. Die Skalierung von der Produktion bis in die Entwicklung und das Testing sind zentral, einerseits für eine effiziente Entwicklung und andererseits für eine lokale Nachvollziehbarkeit des Systems durch den

Entwickler. Zusammen mit Unveränderbarkeit und Reproduzierbarkeit erlauben diese Faktoren nicht nur, die Kontrolle und die Übersichtlichkeit zu wahren, sondern – verglichen mit traditionellen Lösungen – auch viele Probleme zu vermeiden.

### Durchgängigkeit

Erfolgreicher Code ist einfacher Code: Textdateien, die sowohl Mensch als auch Maschine lesen und verarbeiten können. Code lässt sich nicht nur für Implementation und Deployment verwenden, er kann auch in die Analyse und die Konzeption einfließen. Dies schafft eine Durchgängigkeit über den ganzen Produkt-Lebenszyklus.

Aus diesem Grund ist der Business-Analyst heute aus grösseren IT-Projekten nicht mehr wegzudenken. Er weiss, was es schon gibt, er erhebt die (neuen) Anforderungen, konzipiert und dokumentiert eine Lösung. Diese Aufgaben bleiben grundsätzlich gleich. Die Vorgehensweise und die Art des Outputs ändern sich jedoch wesentlich, denn «klassische» Ansätze sind mit Medienbrüchen verbunden. Diese Medienbrüche verhindern das schnelle und korrekte Informationsmanagement, das unerlässlich ist, um die heutigen Ansprüche an Effizienz und Automation zu erfüllen.

## ERFOLGREICHER CODE IST EINFACHER CODE: TEXTDATEIEN, DIE SOWOHL MENSCH ALS AUCH MASCHINE LESEN KÖNNEN.

Der entscheidende Schritt besteht nun darin, das Konzept und das Informationsmodell der Business-Analyse (mit Use Cases, User Interfaces usw.) als Datenmodell zu speichern. Ein solches Modell existiert bereits – zumindest in den Köpfen der Business-Analysten. Seine Vorteile sprechen für sich:

- Die Eingabe der Informationen ist strukturiert: Jede Information gehört an einen bestimmten Ort und wird nicht redundant geführt.
- Die erfasste Information ist aktuell und sofort im Modell verfügbar: Wer Zugriff auf gewisse Daten will, sei es der Kunde, das Software-Team oder der Betrieb, kann diese jederzeit in der gewünschten Form erhalten. Zusätzlich fliesst Information in Form von Kommentaren und Feedback sofort zurück.
- Beziehungen/Verknüpfungen zwischen den Konzepten können erstellt und verfolgt werden, was zu einem stimmigen Gesamtbild des Outputs beiträgt.
- Für Weiterentwicklungen kann auf vorhandene Informationen zurückgegriffen werden. Zudem ist es möglich, Änderungen an diesen Informationen zu erfassen und durch die Verknüpfungen Lücken und unerwünschte Auswirkungen zu vermeiden. Das Informationsmodell bildet das Fundament für die Projektplanung und -methodik: Seine Konzepte lassen sich als Tasks

definieren. Durch Verknüpfung können alle nötigen Informationen für einen Task eingeholt werden. Dies ermöglicht eine bessere Aufwandsschätzung und somit eine bessere Verfolgung des Vorhabens. Nützlich sind dabei etwa Cockpits, die eine Übersicht über den Fortschritt der Entwicklung geben. Diese Informationen sind projekttechnisch. Die fachlichen Informationen sind reine Referenzen zum Informationsmodell.

Alternativ ist es möglich, zum Beispiel Domain-Specific Languages (DSL) einzusetzen, um Systeme in Code zu modellieren. Das Modell beschreibt Komponenten sowie deren Schnittstellen und Interaktionen. Es kann in Repositories abgelegt und online zugänglich gemacht werden. Modelle sind hier nicht auf die Beschreibung begrenzt, sondern dienen auch als Ausgangspunkt für die Erstellung von Software-Komponenten. Sie lassen sich aktualisieren oder in späteren Phasen eines Projekts in neuere Modelle überführen. Umgekehrt ermöglichen sie bei Problemen wertvolle Rückschlüsse auf betroffene Use Cases.

### Die Erkenntnis

Die nächsten Schritte und Erfolge des Software Engineering liegen immer noch «in the Code», allerdings mit Schwerpunkt auf Digitalisierung und Automatisierung. Damit gehen neue Herausforderungen einher, insbesondere für Architektur und Design. Für all diese Aufgaben braucht es nach wie vor ausgewiesene Software-Ingenieure, die in der Lage sind, neue Wege zu beschreiten und im Team zu arbeiten. ■

---

### Remo Meier

*Remo Meier, MSc. und Dr. sc. ETH Zürich, arbeitet seit 2013 im Application Engineering von AdNovum. Dieser Bereich beschäftigt sich mit der Bereitstellung von Mitteln für eine effiziente Software-Entwicklung. In diesem Zusammenhang sind Themen rund um Digital Evolution von grosser Bedeutung. Hat Remo Meier mal frei, ist er im Winter öfter in den Bergen anzutreffen.*

### Stefan Ott

*Stefan Ott, MA in Information Management, ist seit 2007 als Expert Business Analyst und Head of Business Analysis bei AdNovum tätig. Innovation und Evolution stehen bei ihm nicht nur in den Projekten auf dem Programm. Er engagiert sich auch inhouse beim Ausbau der Dienstleistungen des Business-Analyse-Teams. Zu seinen Aufgaben gehören die Ausbildung der Mitarbeitenden sowie die Definition und die Weiterentwicklung von Business-Analyse-Prozessen und -Tools. Eines seiner Hobbys ist das Marathonlaufen, während dessen ihm viel Zeit für die Ideenentwicklung und für Optimierungsvarianten bleibt.*

# DIGITALE TRANSFORMATION DURCH BUSINESS ENGINEERING

Die digitale Transformation gilt als Megatrend. Unternehmen kommen nicht umhin, sich für die veränderten Bedingungen fit zu machen. Dabei hilft Business Engineering.

Von Reinhard Jung



Als Schlagwort ist sie in aller Munde: die digitale Transformation. Doch in vielen Unternehmen herrscht nur eine vage Vorstellung davon, was konkret darunter zu verstehen ist. Gewiss greift es zu kurz, wenn eine Bank eine neue E-Banking-Lösung lanciert oder eine Versicherung ihren Kunden ein Online-Portal für Abschluss und Verwaltung von Verträgen zur Verfügung stellt.

## Treiber der digitalen Transformation

Um das «Wesen» der digitalen Transformation zu verstehen, bietet sich zunächst ein Blick auf die Treiber an, die diesen Megatrend ausgelöst haben beziehungsweise begünstigen:

### 1. Technische Vernetzung und Pervasive Computing

Vernetzung und vernetzte Geräte sind inzwischen allgegenwärtig. Vor gerade mal neun Jahren kam das iPhone auf den Markt. Heute sind Smartphones nicht mehr aus dem modernen Alltag wegzudenken. Die Kunden sind nahezu ständig online, sodass sich reale und Online-Welt immer stärker verzahnen.

### 2. Web 2.0, soziale Netzwerke und Communities

Das World Wide Web, ursprünglich eine Quelle für Informationen, hat sich längst zu einer mächtigen Plattform für echte Kommunikation entwickelt (Web 2.0). Aus einer unidirektionalen 1:n-Kommunikation der Unternehmen zu den Kunden hat sich eine m:n-Kommunikation entwickelt. Dadurch kann etwa ein Konsument andere Konsumenten erreichen, wodurch sich die Machtverhältnisse auf vielen Märkten völlig verändern. Es sind soziale Netzwerke und Communities mit ganz unterschiedlichen Zielen entstanden.

## Über Reinhard Jung

Reinhard Jung ist Professor für Business Engineering an der Universität St. Gallen und Direktor des Instituts für Wirtschaftsinformatik. Er ist Akademischer Direktor des Executive MBA HSG in Business Engineering und Executive Diploma HSG in IT Business Management sowie Leiter des Masterprogramms Business Innovation und Vorstandsmitglied von SIRA (Swiss Informatics Research Association). Seine Forschungsschwerpunkte liegen auf dem Business Engineering, der digitalen Transformation und dem Customer Relationship Management. Er nimmt als Referent an nationalen und internationalen Konferenzen zu aktuellen Business-Engineering- und Digitalisierungsthemen teil.

### 3. Inhalte erzeugen und teilen

Die Bereitschaft der «digitalen» Akteure, Inhalte zu erzeugen und mit anderen zu teilen, steigt. Konsumenten beteiligen sich aus zumeist nicht materiellen Motiven an Innovationsprozessen («Prosumer») oder erbringen sogar Services für andere Kunden. Migipedia der Migros ist ein eindrucksvolles Beispiel.

### 4. Differenzierungsdruck

Viele Märkte sind gesättigt und es herrscht ein intensiver Verdrängungswettbewerb. Die Differenzierung über angestammte Produkte und Dienstleistungen wird zunehmend schwieriger. Neue, zumeist digitale Zusatzleistungen ermöglichen Zusatznutzen für den Kunden und damit eine bessere Differenzierung am Markt.

### 5. Erwartungshaltung

Komfortable Online-Services erhöhen die Erwartungen der Kunden an ihre Marktpartner: Suchen soll so einfach sein wie bei Google, Einkaufen so komfortabel wie bei Amazon und die

Bedienung digitaler Produkte so intuitiv wie bei iOS-Geräten. Anbieter, die nicht mithalten können, geraten schnell ins Hintertreffen. Der Wettbewerb ist intensiv und die Kundenloyalität nimmt tendenziell ab.

## 6. Neue Wettbewerber

In vielen Branchen sind heute zwei Trends erkennbar: Zum einen dringen etablierte und teilweise grosse Unternehmen in andere Branchen vor. Amazon etwa startete mit dem Verkauf von Büchern und ist heute eher ein Online-Kaufhaus für den kompletten Haushaltsbedarf. Zum anderen entstehen neue Unternehmen, die gezielt Wertschöpfungsstufen von etablierten Unternehmen adressieren und durch hochinnovative Produkte und Dienstleistungen substituieren. Ein bekanntes Beispiel ist WhatsApp, das quasi über Nacht den Kurznachrichtenservice der grossen Telekomunternehmen (SMS) «kannibalisierte» und bei diesen massive Umsatzeinbrüche verursachte.

**START-UPS HABEN EINEN  
ENTSCHEIDENDEN VORTEIL:  
SIE MÜSSEN KEINE VORHANDENEN  
STRUKTUREN, PRODUKTE USW.  
INFRAGE STELLEN.**

### Das «Wesen» der digitalen Transformation

Unternehmen, die konsequent auf diese Treiber reagieren und sich durch eine digitale Transformation für die veränderten Bedingungen fit machen, tun dies meist ganzheitlich in folgenden Handlungsfeldern:

1. Geschäftsmodell: Neue Produkte und Dienstleistungen sind gemäss veränderter Kundenbedürfnisse zu definieren, bestehende durch Zusatzleistungen zu ergänzen (hybride Wertschöpfung).
2. Customer Experience: Ausgehend von Kundenbedürfnissen oder sogar -prozessen («Customer Journey») wird das Kundenerlebnis gezielt neu gestaltet.
3. Wertschöpfungsprozess: Dieser Prozess wird so weit wie möglich digitalisiert beziehungsweise automatisiert. Die Produktion spricht in diesem Kontext von Industrie 4.0.

Interessant ist, dass Start-ups prinzipiell identisch vorgehen. Sie haben jedoch einen entscheidenden Vorteil: Sie müssen keine vorhandenen Strukturen, Produkte usw. infrage stellen. Dieser «Ballast» führt in etablierten Unternehmen zu erheblichem Widerstand gegen die Transformation.

### Business Engineering als Strukturierungsmodell

Das an der Universität St. Gallen entwickelte Konzept des Business Engineering kann als Rahmen für die digitale Transformation dienen. Als umfassender Ansatz für die Veränderung zeigt

es den systematischen Weg durch die Transformation für alle Gestaltungsebenen des Unternehmens auf. Ganz wichtig sind dabei die Leadership-Aspekte, die so oft über Erfolg oder Misserfolg entscheiden. Business Engineering hilft, den Veränderungsprozess für alle Mitarbeitenden transparent und nachvollziehbar zu machen. Für «Veränderungsmanager» stellt es daher einen umfangreichen Werkzeugkasten bereit.

Business Engineering fusst auf einem konsequenten Outside-in-Denken, von den Kundenbedürfnissen ausgehend und nicht vom Produktionsprozess, und stellt Methoden und Modelle zum systematischen und gestaltenden Umgang mit dem Wandel bereit. Als Top-down-Ansatz integriert es die diversen Handlungsfelder eines ganzheitlichen Wandels. Im Rahmen der digitalen Transformation mag es beispielsweise bei folgenden Aktivitäten unterstützen:

**Geschäftsstrategie:** Erweiterung und Neubewertung des Geschäftsmodells und Ergänzung des Produktportfolios durch beziehungsrelevante (digitale) Services.

**Geschäftsprozesse:** Nutzung neuer/zusätzlicher Kommunikationskanäle mit den Kunden und Umgestaltung der Kundenservice- und Kommunikationsprozesse, um mehr Kundennutzen zu generieren und zugleich umfassendes Wissen über Kunden(bedürfnisse) aufzubauen.

So kann das Business Engineering Framework mit seinem systematischen Vorgehen helfen, Veränderungsprojekte für die digitale Transformation erfolgreich umzusetzen.

Weitere Informationen zum Executive MBA HSG in Business Engineering: [www.embe.unisg.ch](http://www.embe.unisg.ch) ■

---

## Impressum

### Herausgeber:

AdNovum Informatik AG  
Corporate Communication  
Röntgenstrasse 22  
8005 Zürich  
Telefon +41 44 272 6111  
E-Mail [info@adnovum.ch](mailto:info@adnovum.ch)  
[www.adnovum.ch](http://www.adnovum.ch)

### Verantwortung und Redaktion:

Andrea Duttwiler  
Feedback: [notitia@adnovum.ch](mailto:notitia@adnovum.ch)

### Gestaltung und Realisation:

Comuniq, Zürich

### Fotografie:

Getty Images (Titel), Gerry Nitsch, Zürich  
Gedruckt auf Balance Pure





**WIE VIELE IDEEN SIE AUCH HABEN.  
WIR ENTWICKELN DIE SOFTWARE DAZU.**

MIT ADN OVUM ALS SOFTWARE-PARTNER ERLANGT IHRE BUSINESS-IDEE SCHNELL DIE NÖTIGE MARKTREIFE. WIR UNTERSTÜTZEN SIE VON DER IDEENFINDUNG BIS ZUR PRODUKTLANCIERUNG UND MACHEN SIE BEI DER MARKTEINFÜHRUNG ZUM FIRST MOVER. NEHMEN SIE FRÜHZEITIG MIT UNS KONTAKT AUF - SELBST WENN DIE BIG IDEA NOCH NICHT SO BIG IST. ADN OVUM INFORMATIK AG, RÖNTGENSTRASSE 22, 8005 ZÜRICH, SCHWEIZ, TELEFON +41 44 272 61 11, WWW.ADN OVUM.CH

HIGH-END SOFTWARE & SECURITY ENGINEERING.

**ADNOVUM**