NOTITIA

AdNovum

Bemerkenswertes von und über AdNovum

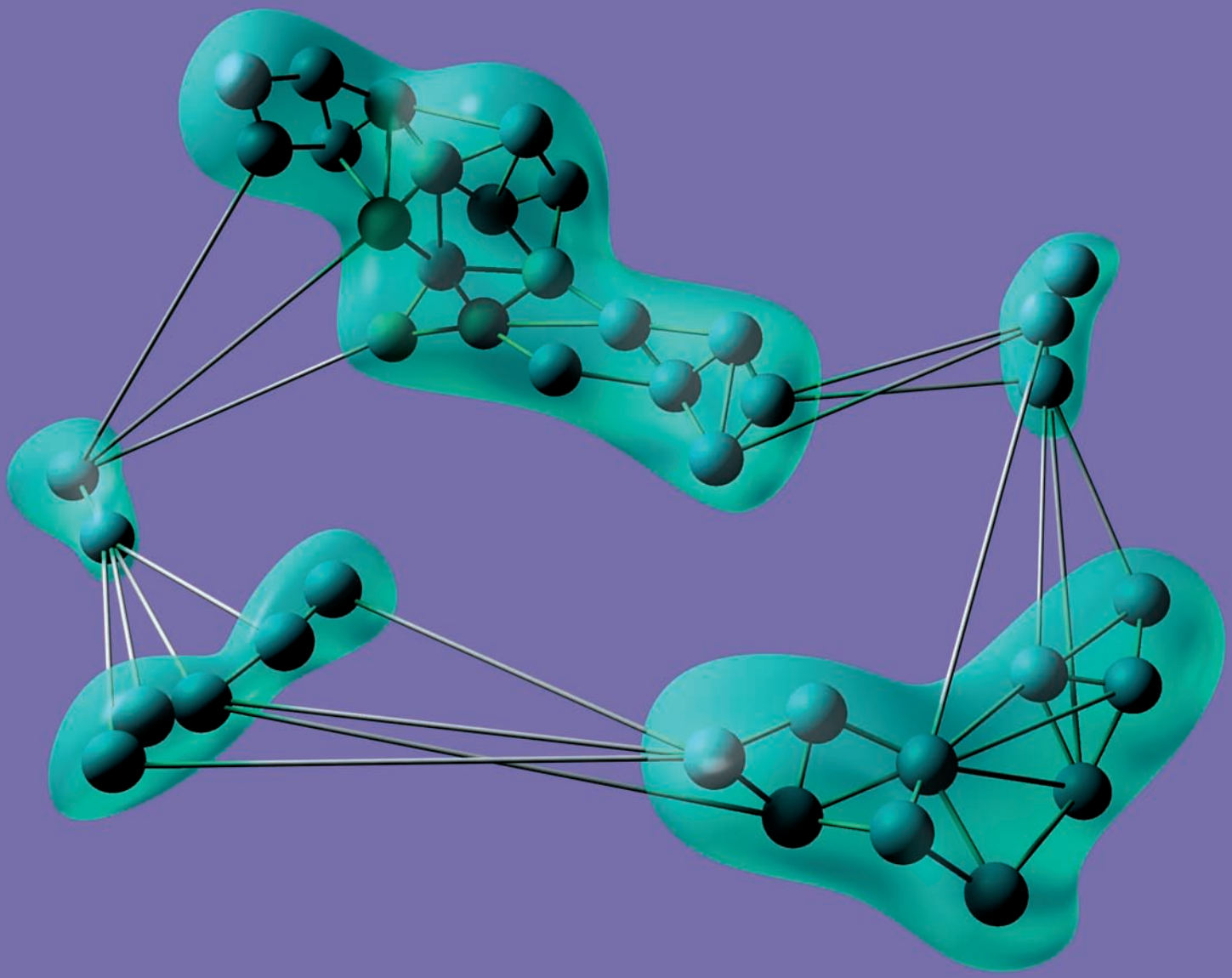
Visuelle Clusteranalyse
Informationsgewinnung für schnelle und fundierte Entscheide

Qualitätssicherung
Ein interaktiver Prozess von der Codierung zur Installation

Identity Management
Realisierung und Einführung massgeschneiderter IDM-Systeme

Herbst 2005, Nr. 9

Clusteranalyse und Qualitätsmanagement



Liebe Leserin, lieber Leser

Clusteranalyse – ein bewährtes Verfahren zur Mustererkennung in Datensätzen. Unter Beizug neuer Visualisierungs- und Interaktionsparadigmen lässt sich damit in grossen, vieldimensionalen, lose besetzten Datenräumen die Komplexität reduzieren und handliche Information gewinnen. Bislang wurde die Clusteranalyse vor allem auf Informationssysteme angewendet, zum Beispiel beim Data Warehousing oder zur Beurteilung von Finanzinstrumenten. Auch im IT-Umfeld existieren jedoch hochkomplexe Zusammenhänge, die den Einsatz solcher leistungsfähiger Werkzeuge lohnend machen, insbesondere im Hinblick auf die gestiegenen Sicherheitsanforde-

Visuelle Clusteranalyse

Systeme zur Datenverwaltung (Data Warehouses, Finanzdaten) stellen heute ein gewaltiges Informationspotential dar, dessen Nutzung jedoch leistungsfähige Werkzeuge erfordert. Insbesondere die moderne Informationsvisualisierung bietet effiziente Verfahren zur Analyse solcher hochdimensionaler, grosser Datenvolumen.

Von Tom Sprenger

In modernen Systemen sind das Volumen und die Komplexität der anfallenden Daten über die letzten Jahre um Grössenordnungen gewachsen. Aus dieser Datenschwemme innert nützlicher Frist wertvolle Informationen zu extrahieren, diese zu interpretieren und basierend darauf die richtigen Entscheide zu treffen, wird in verschiedensten Bereichen zu einer anspruchsvollen Herausforderung (siehe Kasten); ging es früher noch um die Suche nach der Nadel im Heuhaufen, so stellt sich heute oft die Frage, in welchem der vielen Heuhaufen man überhaupt zu suchen beginnen soll.

Zwar existieren teilweise ausgefeilte domainspezifische Analysewerkzeuge. Wenn es aber darum geht, gefundene Informationen und die daraus gewonnenen Erkenntnisse zu präsentieren, so bleibt es meistens bei tabellarischen Repräsentationen oder einfachen Grafiken (Linien-, Kuchen- oder Bar-Charts). Obwohl viele Analysten es gewohnt sind, mit Tabellen und Diagrammen zu arbeiten, ist diese Art der Darstellung für heutige Datenvolumen eher ungeeignet. Solche starren zweidimensionalen Darstellungen leiden an zwei wesentlichen Problemen. Zum einen kann damit jeweils nur ein sehr beschränkter Ausschnitt der Gesamtinformation dargestellt werden, ohne die Präsentation zu überladen. Zum anderen sind solche Darstellungen inhärent zu wenig mächtig, um die in den Daten vorhandenen Informationen und komplexen Zusammenhänge abzubilden und für Entscheidungsprozesse verfügbar zu machen. Als Konsequenz bleiben potentiell wichtige Informationen unentdeckt und ungenutzt. Im Weiteren funktionieren die meisten dieser Tools im so genannten Batch Mode, der Benutzerinteraktionen während des Analyseprozesses per se verhindert. Weil die Eingangsdaten signifikant variieren können, ist eine für alle Fälle korrekte Parametrierung der involvierten Algorithmen und der Darstellung praktisch unmöglich. Als Konsequenz sind typischerweise mehrere Durchläufe notwendig, um ein brauchbares Resultat zu erhalten; ein Umstand, der diese Verfahren in einer Zeit schneller Entscheide nicht mehr opportun erscheinen lässt.

Es sind Systeme und Verfahren gefragt, die dem grossen Volumen, der Komplexität und der

Potentielle Anwendungsgebiete visueller Analyseverfahren

- Finanzdaten
- grosse Dokumentsammlungen
- Qualitätssicherung grosser Software-Projekt-Repositories
- Abhängigkeiten in grossen, verteilten Systemen
- Security-relevante Daten und Systemkonfigurationen
- Soziale Strukturen in Organisationen
- Product Information Management
- Kreditrisikobeurteilungen
- Fraud Detection

rungen: Beispielsweise können Berechtigungsprofile nach Mustern durchsucht und damit Eigenheiten von Profiltypen erkannt und Verletzungen von Zugriffsbestimmungen aufgespürt werden. Mehr darüber erfahren Sie im einleitenden Artikel von Tom Sprenger.

Vielschichtig und komplex ist auch das Daily Business der AdNovum, die zielführende und termingerechte Realisierung von Software-Projekten. Auch hier lassen sich jedoch wiederkehrende Muster erkennen, und AdNovum nutzt diese zur Standardisierung und Automatisierung ihrer Prozesse, um damit effizienter und in deterministischer Qualität Software zu produzieren und Projekte durchzuführen.

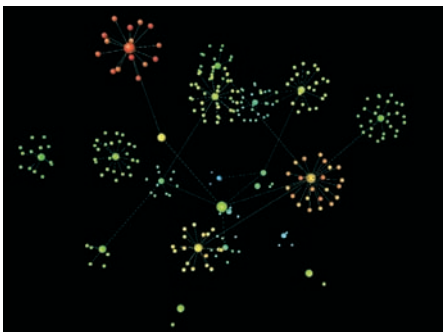
Um die automatisierte Qualitätssicherung als Dreh- und Angelpunkt in diesem Vorhaben geht es im Interview. Domenico Bernardo als Verantwortlicher für das Release Engineering und Daniel Spörndli als Integrator in einem grossen Portal-Projekt stehen Rede und Antwort und zeigen auf, wie sich QS in einem AdNovum-Projekt konkret manifestiert.

Lösungen rund um IDM werden seit geraumer Zeit von diversen grösseren Anbietern stark beworben, was auf ein entsprechendes Marktpotential hinweist. Im Hintergrundartikel zeigt Andreas Petralia auf, worin der Aufwand besteht, ein IDM-System bei einem grossen Finanzdienstleister einzuführen, und dass die integrativen und organisatorischen Aspekte als Key Success Factors das Hauptaugenmerk verdienen.

Zum Schluss darf ich noch mit Freude darauf hinweisen, dass unser höchst erfolgreich agierender Partner Finnova sich und seine Lösung in dieser Ausgabe vorstellt.

Stefan Arn

CEO AdNovum Informatik AG



Dynamik gewachsen sind und dem Benutzer die interaktive Analyse der Daten erlauben – Verfahren also, die mit mehr als nur zwei oder drei Dimensionen gleichzeitig umgehen können und diese höherdimensionalen, abstrakten Datenräume auf einem Bildschirm oder einem Stück Papier konkret und aussagekräftig darzustellen vermögen. Dabei müssen sie eine gewisse Unschärfe in der Suche unterstützen, denn nicht immer ist bei der Analyse komplexer Datenvolumen von Beginn weg klar, was genau gesucht wird. Der Prozess – auch explorative Analyse genannt – ist häufig vielmehr eine interaktive, meist ungerichtete Suche, fast ein eigentliches Stöbern, nach Mustern, Trends und Singularitäten in den Daten. Erstes Ziel ist es, anhand der Daten eine Hypothese über Eigenschaften und Informationen aufzustellen, die dann im weiteren Prozess – der so genannten bestätigenden Analyse – zielstrebig erhärtet bzw. verworfen wird.

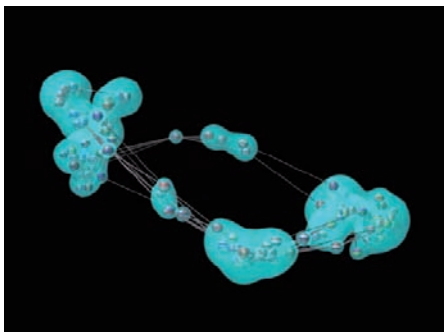
Use Vision to Think

Nach dem Motto «Ein Bild sagt mehr als tausend Worte» nehmen sich neue Verfahren und Tools aus der noch relativ jungen Domäne der Informationsvisualisierung dieser Problematik an. Die Schlüsselidee dabei ist, den Benutzer mit seinem überaus leistungsfähigen kognitiven System und seiner Fähigkeit zum vernetzten Denken als festen Bestandteil mit in den Prozess der Informationsextraktion und Wissensgenerierung zu integrieren. Abstrakte Datenobjekte und deren Attribute werden mittels geeigneter Paradigmen in visuelle Repräsentationen abgebildet, die es auf einem qualitativen Level erlauben, explorativ durch hochdimensionale, komplexe Datenräume zu navigieren. Die in der Visualisierung intuitiv beobachtbaren Muster lassen Rückschlüsse auf Eigenschaften der zugrunde liegenden Daten zu. Hauptziele sind das Erkennen von Regionen mit ähnlichen Eigenschaften sowie das Aufspüren von Anomalien und das Darstellen von Beziehungen zwischen verschiedenen Datenobjekten. Solche Verfahren liefern nicht nur eine effektive Art der Informationsverdichtung, sondern machen auch konsequenten Gebrauch von der leistungsfähigen Mensch-Computer-Schnittstelle. Die Informationen können vom Benutzer in einer parallelen Weise aufgenommen werden, während sie gleichzeitig vom stärksten existierenden Mustererkennungsprozessor – dem menschlichen kognitiven System – analysiert werden.

Theorie: Cluster and Conquer

Eine der leistungsfähigsten und wichtigsten Visualisierungstechniken ist die visuelle Clusteranalyse, welche die klassische Clusteranalyse um neue Visualisierungs- und Interaktionsparadigmen erweitert.

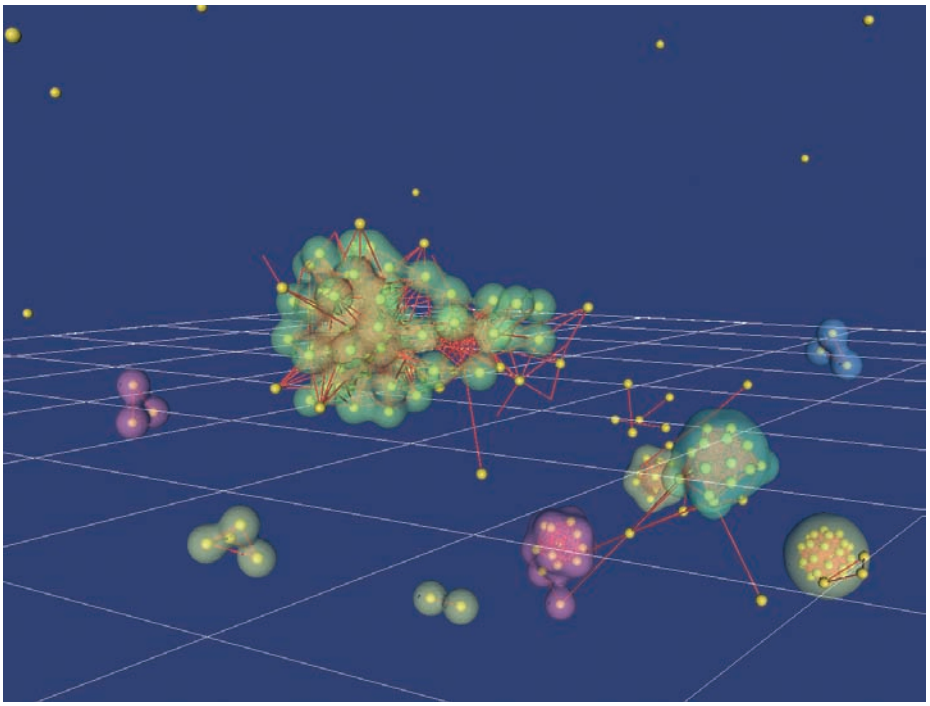
Die visuelle Clusteranalyse stützt sich auf die Erkenntnis, dass in grossen Datenvolumen die einzelnen Datenobjekte an Relevanz verlieren, die Relationen zwischen den Daten hingegen an Bedeutung gewinnen. Die Datenobjekte werden beim Clustering in Gruppen ähnlicher Objekte aufgeteilt und diese Gruppen zu repräsentativen Clusterobjekten zusammengefasst.



Die Grundlage für die visuelle Clusteranalyse bildet die so genannte Distanz-Ähnlichkeits-Metapher, welche die analytische Betrachtung von Daten mit der visuellen Wahrnehmung entsprechender Strukturen

verbindet. Die Grundidee der Metapher ist die definierte Abbildung von Ähnlichkeit im abstrakten Datenraum auf euklidische Distanzen in der Visualisierung. Das Resultat sind Gruppierungen von Datenobjekten mit folgender Semantik: Je näher zwei Objekte in der Visualisierung beieinander liegen, desto grösser ist ihre Ähnlichkeit im Datenraum.

Die Berechnung des räumlichen Layouts der Datenobjekte erfolgt über abstrakte 3D-Graphstrukturen. Die Positionsberechnung beruht im Kern auf der Quantifizierung der Ähnlichkeit von in Beziehung stehenden Objekten, wobei die gefundenen Quantitäten wiederum die Parameter eines Masse-Feder-basierten Layoutsystems konfigurieren. Weil die Federhärten und die berechneten Ähnlichkeitswerte korrespondieren, konvergiert das System in ein energetisches Minimum, das die inhärent in den Daten enthaltenen Beziehungen und Zusammenhänge als räumliche Nachbarschaften widerspiegelt.



Cluster aus Berechtigungsprofilen einer Organisation.

Praxis: Analyse von Berechtigungsprofilen

Das folgende Beispiel zeigt, wie die historisch und unstrukturiert gewachsenen Berechtigungsprofildaten einer Organisation mittels moderner Visualisierungstechniken analysiert, restrukturiert und in einen kontrollierten Betrieb übergeführt werden können. Dabei wird das Ziel angestrebt, die Überschaubarkeit zu erhöhen, indem der Profilwildwuchs eingedämmt und durch ein definiertes Set von Grundprofilen ersetzt wird. In einer Organisation mit begrenzt vielen Rollen sollte es auch nur begrenzt viele Berechtigungsprofile geben. Es soll also nicht jedem Benutzer ein eigenes Berechtigungsprofil zugewiesen werden, sondern aufgrund seiner Rolle(n) eines oder mehrere der Grundprofile aus einem definierten Set.

Die Motivation liegt zum einen in der Qualitätsverbesserung im Bereich der Sicherheit durch ein gezieltes Einschränken der Varianz. Zum anderen wird durch die Strukturierung auch die Betreibbarkeit des Systems verbessert, was längerfristig zu entsprechenden Kosteneinsparungen führt.

Die zu analysierenden Daten bestehen im Beispiel aus 591 Berechtigungsprofilen. Jedes Profil wiederum setzt sich aus einem 69-dimensionalen Berechtigungsvektor zusammen. Jede Dimension entspricht einer bestimmten Berechtigung, wobei sich die Berechtigung wiederum aus vier frei kombinierbaren Zugriffsrechten (erstellen, ändern, lesen und löschen) zusammensetzt.

A priori hat man keine Informationen über die Eigenschaften der zu analysierenden Profile. Um eine erste Übersicht zu erhalten, werden die Berechtigungsprofile so, wie sie heute definiert sind, mittels der Clustervisualisierung analysiert. Dazu wird eine geeignete Metrik zur Quantifizierung der Ähnlichkeit zwischen zwei Berechtigungsvektoren definiert. Die Visualisierung wird mit den Profildaten und der Metrik konfiguriert und gestartet. Aufgrund der resultierenden Grafik kann in einem ersten Schritt entschieden werden, ob das geplante Vorhaben überhaupt realistisch ist. Es kann sich zum Beispiel zeigen, dass die Berechtigungsprofile dermassen chaotisch sind, dass in den Daten keine Cluster zu erkennen sind. In einem solchen Fall muss mit der Strukturierung der Profile auf der grünen Wiese begonnen werden. Im anderen, im aktuellen Kontext interessanteren Fall kristallisieren sich in der Visualisierung klare Gruppen – sprich Cluster – von Berechtigungsprofilen mit ähnlichen Eigenschaften heraus. In der Abbildung (siehe obiges Beispiel) werden die gefundenen Cluster vom Visualisierungssystem mit semitransparenten umhüllenden Flächen markiert. Die so erhaltenen Cluster geben Aufschluss über Anzahl und Art der benötigten Grundprofile und können somit als Basis für deren Definition verwendet werden. Mittels eines Drill-down auf einem Clusterobjekt (Anzeige spezifisch detaillierter Information) kann genauere Auskunft über den mittleren Berechtigungsvektor respektive das typische, mittlere Berechtigungsprofil dieser Gruppe erlangt werden.

> A priori hat man keine Informationen über die Eigenschaften der zu analysierenden Daten.

Nun liegen aber noch lange nicht alle Berechtigungsprofile in einem Cluster. In der Visualisierung sind viele zwischen den Clustern liegende Objekte zu erkennen. Die Interpretation dieser Beobachtung ist relativ einfach. Objekte kommen zwischen zwei Cluster zu liegen, wenn sie sowohl zum einen wie auch zum anderen Cluster eine gewisse Affinität haben. Das heisst nichts anderes, als dass sich solche Berechtigungsprofile aus einer Überlagerung mehrerer Grundprofile zusammensetzen.

Die Visualisierung erlaubt weitere Rückschlüsse auf die Daten. Man hat vielleicht bereits die verschiedenen Ausreisser am Rande

der Grafik bemerkt: Neben den Clustern der Grundprofile erkennen wir nun also auch Anomalien in den Berechtigungsprofilen. Bei solchen «Sonderfällen» ist die kritische Frage angebracht, was der genaue Sinn und Zweck einer Profildefinition ganz an der Peripherie der Visualisierung ist respektive ob es im konkreten Einzelfall überhaupt Sinn macht, ein Profil zu definieren: Wem wurde ein solches Profil zugewiesen und warum? Wird diese Art der Analyse kontinuierlich ausgeführt, können Fehler und Risiken in komplexen Security-Konfigurationen rasch entdeckt werden. Mit der Elimination solcher Ausreisser und Fehlkonfigurationen lässt sich die Konsistenz und Qualität der Daten und damit im konkreten Beispiel die Security des Systems erhöhen.

Ausblick

Moderne Visualisierungsmethoden machen auch grosse und komplexe Datenvolumen wieder nutzbar. Sie unterstützen dabei insbesondere die explorative Datenanalyse, bei der initial noch nichts über die Eigenschaften der Daten bekannt ist. Mit visuellen Clusteringverfahren kann die initiale Komplexität auf ein beherrschbares Niveau heruntergebrochen werden, und Datencharakteristiken lassen sich so auf einem qualitativen Level schnell kommunizieren. Eine spezielle Eigenschaft dieses Ansatzes ist, dass Cluster ähnlicher Daten sowie Anomalien effizient erkannt und interpretiert werden können. Die erzeugten interaktiven Grafiken beschleunigen den Prozess, aus den Daten Informationen zu extrahieren und daraus Wissen aufzubauen. Dies wiederum ermöglicht erst die heute benötigten schnellen und fundierten Entscheide der verantwortlichen Personen.

Die vorgestellten Verfahren sind mehrheitlich datenneutral und lassen sich somit universell einsetzen: Die möglichen Anwendungen reichen von der Analyse grosser objektorientierter Programme zwecks Architektur- und Qualitätssicherung über das Visualisieren von Laufzeitzusammenhängen in komplexen Serviceinfrastrukturen oder die Analyse sozialer Strukturen in einer Gesellschaft oder Firma bis hin zum Detektieren auffälliger Bewegungen, Operationen oder Transaktionen in komplexen Bankensystemen. Gute Resultate wurden auch mit der Visualisierung von Resultaten aus Suchmaschinen erzielt, welche die feingranularere Affinität zwischen den gefundenen Dokumenten nicht wie bei der bekannten Trefferliste ignoriert, sondern eine Menge von Dokumentclustern an den Benutzer weitergibt.

Die mit den Visualisierungsverfahren gewonnenen Repräsentationen stellen eine komprimierte Sicht auf die real vorhandenen Daten dar. Wegen ihrer Informationsdichte sind solche Grafiken jedoch ohne das entsprechende Kontextwissen nur schwer zu interpretieren. Die Annahme, dass visuelle Verfahren ein fundiertes Fachwissen obsolet machen, ist somit wie beim Einsatz der meisten anderen Werkzeuge ein Trugschluss. Vielmehr ist es das Ziel, den Experten stärker in den Analyseprozess einzubeziehen. Die Medienbruchstelle Mensch-Maschine und Maschine-Mensch soll so effektiv wie möglich gestaltet werden, um den Experten in seinen Aufgaben zu unterstützen. Mit der Informationsvisualisierung und dem Einsatz der daraus resultierenden Verfahren ist ein erster Schritt in diese Richtung getan.

> Visuelle Clusteringverfahren ermöglichen die heute benötigten schnellen und fundierten Entscheide.

Tom Sprenger

Tom Sprenger jonglierte im Rahmen seiner Dissertation in der Computer Graphics Group der ETH mit Informationsvisualisierung in komplexen Datenräumen, einem Thema, das er auch in den HP Labs in Palo Alto weiterverfolgte. Seit 2000 für die AdNovum aktiv, übernahm er eine zentrale Rolle beim Aufbau des Java Engineering und des Testing Environment. Aktuell ist er damit beschäftigt, in der AdNovum den strategischen Bereich Quality Assurance Engineering auszubauen. In der Freizeit kurvt Tom Sprenger gerne auf seinem Segway durch die Gegend.

Qualitätssicherung

Notitia unterhielt sich mit Daniel Spörndli und Domenico Bernardo über die Qualitätssicherungsmassnahmen und -strategien der AdNovum in der Software-Entwicklung und im Release Engineering.

Interview: Manuel Ott

NOTITIA: Ist Qualitätssicherung (QS) in der AdNovum ein Thema?

Daniel Spörndli: Selbstverständlich! Wie jedes Unternehmen, das Qualitätsprodukte herstellt, haben auch wir aktiv Massnahmen zur Sicherstellung der Qualität ergriffen. QS ist in der AdNovum von strategischer Bedeutung. Wir Entwickler werden in den verschiedenen Aspekten der QS durch das Quality Assurance Engineering Team aktiv unterstützt. Zudem erfolgt die AdNovum Software-Produktion entlang einem genau definierten Prozess. Innerhalb dieses Prozesses haben wir umfangreiche Vorkehrungen getroffen, um die Qualität über alle Stadien eines Projekts sicherzustellen. Die QS beginnt bei automatisierten Unit-Tests in der frühen Entwicklungsphase und führt über Integrations- und Lasttests bis hin zu manuellem Testing durch ein separates Team kurz vor der Auslieferung.

Domenico Bernardo: QS ist auch im Release Engineering wichtig. Die Massnahmen sind vergleichsweise einfach, aber sehr effektiv. Zum Beispiel installieren wir die Software ab dem ersten Prototypen via Packages. Packages sind eigenständige Auslieferungseinheiten unserer Software, die eine einfache Installation erlauben. Durch die Installation in frühem Stadium stellen wir sicher, dass die Packages alle Files installieren und dass beim Zusammenstellen nichts vergessen wurde.

Hinzu kommt, dass wir die Software vor einer Lieferung immer auf einen so genannten Integrationsserver installieren. Auf den Integrationsserver hat nur der zuständige Release Engineer Zugriff; Entwickler können darauf keine Manipulationen vornehmen. Wenn eine Software nicht funktioniert, kann somit nicht direkt auf dem Server gepatcht werden, sondern es ist eine Neuinstallation des Package notwendig. Damit erzwingen wir den gleichen Ablauf, wie wir ihn im produktiven Umfeld beim Kunden vorfinden.

> « Der Zusatzaufwand für QS entsteht in einer frühen Projektphase, zahlt sich jedoch später aus. »

Welchen Stellenwert geniesst die QS in der AdNovum bzw. was ist die Motivation für den Zusatzaufwand?

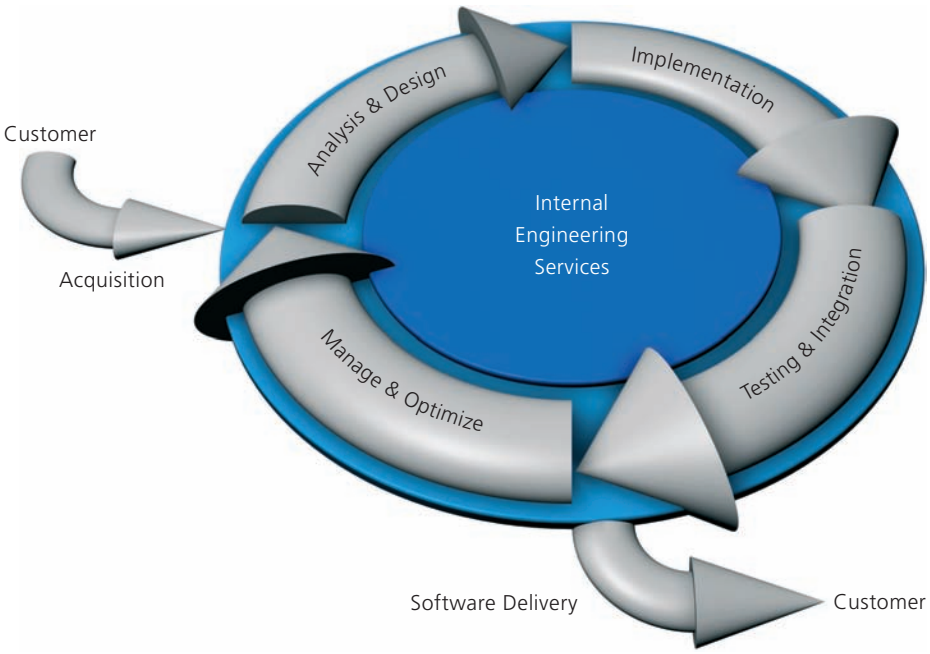
Daniel Spörndli: Die umfangreichen Massnahmen für QS generieren in der Tat einen Zusatzaufwand. Allerdings gewinnen wir durch diesen Zusatzaufwand die Sicherheit, eine hohe Qualität in jedem Release gewährleisten zu können. Der Zusatzaufwand entsteht in einer frühen Projektphase und zahlt sich in den späteren Phasen und in einer besseren Betreibbarkeit aus. Der Aufwand besteht grundsätzlich darin, dass nebst funktionalem Code auch Testcode geschrieben werden muss. Ausserdem ist ein Projekt so zu gestalten, dass automatisierte Tests ausgeführt werden können. Dabei geht es nicht nur um automatisierte Unit-Tests, sondern auch um automatisierte Integrationstests, wobei idealerweise das ganze Environment für den Test automatisch aufgesetzt und danach wieder gelöscht wird.

Domenico Bernardo: Als Release Engineer bin ich auch an den Installationen beteiligt. Spätestens bei der Installation beim Kunden bin ich selber von der Qualität der Packages betroffen. Für uns ist dies Motivation genug, um auch aus Sicht des Release Engineering hohe Ansprüche an die Qualität der Packages zu

stellen. Es gibt wenig Peinlicheres, als wenn beim Kunden die Installation wegen eines Fehlers im Package nicht abgeschlossen werden kann.

Sie haben automatisierte Tests angesprochen; lässt sich mit automatisierten Tests alles abdecken?

Daniel Spörndli: Ideal wäre es natürlich schon, wenn man eine vollständige Abdeckung durch automatisierte Tests hätte. Aber der Aufwand dafür ist einfach zu gross. In der Praxis muss deshalb in jedem Projekt individuell entschieden werden, welche Bereiche, Komponenten und Funktionen mit automatisierten Tests abgedeckt werden sollen. Auch lassen sich gewisse Dinge gar nicht automatisiert testen: beispielsweise die grafische Benutzeroberfläche einer Web-Applikation in verschiedenen Browsern oder Testszenarios, die nur mit sehr hohem Aufwand automatisiert werden können. Es gilt zudem zu bedenken, dass auch der Testcode sowie die Testdaten während der ganzen Lebensdauer des Projekts gepflegt sein wollen. Deshalb muss für automatische Tests ein fixes Set von Tools bereitgestellt werden. Die Tools ermöglichen es den Entwicklern, effizient automatisierte Tests zu erstellen. Dies soll dann uniform über alle Projekte geschehen.



Software-Engineering-Prozess: Bei jedem Durchlauf werden Fehler eliminiert und damit die Qualität erhöht.

> «Mit standardisiertem Testing können Entwickler von Beginn weg auf den Inhalt fokussieren.»

Das Stichwort Aufwand ist bereits gefallen ...

Daniel Spörndli: Fehler, die erst sehr spät gefunden werden (im schlimmsten Fall erst im produktiv laufenden System), sind am aufwändigsten zu beheben und somit die teuersten. Es ist tatsächlich so, dass sich am Ende die Investition in automatisierte Tests immer auszahlt. Bei späteren Änderungen an einer Software stellen die automatischen Tests sicher, dass nicht neue Fehler eingebaut werden.

Um die Kosten im Griff zu behalten, muss das Testing aber in allen Projekten gleich gehandhabt werden. In der AdNovum wird ein definierter Werkzeugkasten fürs Testing verwendet. Dieser ist integraler Bestandteil der Projekt-Blueprints, d.h. des generischen Grundgerüsts für alle Projekte. Jedes Projekt enthält somit automatisch dieselbe Teststruktur. Damit wird erreicht, dass beim automatisierten Testing der Fokus von Beginn weg auf dem «Was» und nicht auf dem «Wie» liegt.

Gerade in diesem Bereich besteht jedoch auch Verbesserungspotential. Momentan verwendet ein Entwickler noch zu viel Zeit mit dem Setup der Testumgebung. Ziel muss es sein, diesen Aufwand zugunsten des eigentlichen Schreibens von Testfällen zu reduzieren.

Man hört überall von automatisiertem Testing und was in diesem Bereich alles möglich wäre. Wie weit ist die AdNovum konkret?

Daniel Spörndli: Seit geraumer Zeit führen wir in der AdNovum Nightly Builds durch. Die Projekte werden jeweils in der Nacht automatisch aus dem Versionierungssystem ausgecheckt und gebaut. Anschliessend werden die konfigurierten projektspezifischen Tests durchgeführt. Danach werden noch einige projektunabhängige Tests vorgenommen, beispielsweise wird der Code auf «Antipatterns» untersucht. In den Nevis-Middleware-Projekten wird sogar völlig automatisiert ein Standard-setup aus den diversen Komponenten aufgebaut und damit deren Zusammenspiel getestet. Falls bei einem solchen Nightly Build etwas nicht korrekt funktioniert, wird das Entwicklungsteam automatisch per E-Mail darüber informiert.

In Zukunft wollen wir noch weiter gehen und zusätzliche Merkmale überprüfen. So sollen beispielsweise die transitiven Abhängigkeiten zu anderen Projekten überprüft oder ein allfälliges Fehlen von Dokumenten wie Spezifikation oder Installationshandbuch festgestellt werden können.

Domenico Bernardo: In den Nightly Builds werden auch die Packages getestet: Läuft der Build-Prozess fehlerfrei durch, ist das Package installierbar, kann man es aktivieren, lassen sich die zum Package gehörenden Administrations-Kommandos ausführen? Dies sind die wichtigsten Kriterien für ein Package, alles andere muss man bei der Installation auf dem Integrationsserver testen. Sobald die Applikation testbar ist, sind die Packages in Ordnung.

Wie wird sichergestellt, dass die geplanten Funktionen und Fixes in der Lieferung enthalten sind und funktionieren?

Domenico Bernardo: Der Release-Engineering-Prozess ist folgendermassen definiert: Mit dem Business-Projektleiter und dem Technischen Projektleiter wird ein so genannter Commit-Stop vereinbart, ein Termin, der bei uns sehr genau eingehalten wird. Bis zu diesem Termin können die Entwickler die beanstandeten Bugs beheben und geforderte Change Requests (CRs) einbauen. Die Änderungen, die der Entwickler am Code vornimmt, trägt er in einen Change Log ein. Damit ist nachvollziehbar, welche Fixes und Änderungen im Code enthalten sind und welche nicht. Dann kommt

der Release Engineer ins Spiel. Er veranlasst einen Check-out der Software, und anschliessend kompiliert er sie. Treten dabei Fehlermeldungen auf, so muss der Code mit dem Auftrag zum Fix dem Entwickler zurückgegeben und ein neuer Commit-Stop vereinbart werden. Nach erfolgreicher Kompilation werden die Packages gebaut und auf dem Integrationsserver installiert. Sobald die Applikation verfügbar ist, führt das Testing-Team anhand entsprechender Drehbücher die manuellen Tests durch. In der gleichen Phase kontrolliert das Testing-Team auch alle gemeldeten Bugs und CRs anhand der entsprechenden Einträge im Bugzilla. Dieses Open-Source-Produkt wird bei uns als Bug- und Task-tracking Tool eingesetzt. Die Resultate der Tests werden ebenfalls im Bugzilla festgehalten und kommuniziert. Werden dabei alle geforderten Punkte erfüllt und treten keine Fehler auf, kann die Software zur Lieferung freigegeben werden.

> «Die Software wird erst nach umfangreichen Tests und Kontrollen zur Lieferung freigegeben.»

Das Ganze ist vergleichbar mit der Produktion eines Autos: Wenn das Fahrzeug vom Fliessband rollt, kommt der Prüfer, öffnet und schliesst alle Türen und fährt eine gewisse Strecke, um die Bremsen zu testen usw. Wenn beim Test ein Fehler auftritt, muss der Wagen zurück in die Verarbeitung. Wenn aber alles so ist wie gewünscht, findet die Lieferung statt.

Wie findet die Lieferung statt?

Domenico Bernardo: Der Release Engineer wartet die Freigabe durch das Testing-Team ab und nimmt anschliessend anhand eines Delivery Tool die Lieferung vor. Im Delivery Tool sind alle Projekte und Kunden mit den nötigen Informationen eingetragen. Beispielsweise ist darin ersichtlich, über welchen Kanal die Software vom Kunden bezogen wird, welche Personen benachrichtigt werden und welche Versionen bereits geliefert wurden.

Der Release Engineer ist zudem für das Installationshandbuch verantwortlich. Wie die Release Notes ist auch das Installationshandbuch immer Bestandteil der Software-Lieferung. Wenn der Release Engineer alle Komponenten beisammenhat, kann er die Lieferung auslösen. Im Delivery Tool wird jede ausgelöste Lieferung gespeichert, womit jederzeit nachvollziehbar ist, welcher Kunde wann welche Software erhalten hat.

Was geschieht nach der Lieferung?

Domenico Bernardo: Natürlich ist der Prozess für den Release Engineer nach der Lieferung noch längst nicht abgeschlossen: Die Software wird immer in Begleitung eines Release Engineer anhand des Installationshandbuchs installiert. Der Kunde testet anschliessend die Applikation und kann allfällig auftretende Fehler im Bugtracking Tool erfassen. Das Bugtracking Tool, eine vereinfachte Benutzeroberfläche des Bugzilla, stellen wir unseren Kunden über die speziell gesicherte Customer Zone unserer Website zur Verfügung. Dort kann der Kunde die von ihm gemeldeten Bugs oder CRs laufend verfolgen, überprüfen und gegebenenfalls auf «erledigt» setzen. Damit schliesst sich der Kreis der Qualitätssicherung im Software-Engineering-Prozess.

Daniel Spörndli

Daniel Spörndli, diplomierter Informatik-Ingenieur ETH, arbeitet seit bald vier Jahren bei der AdNovum als Applikationsentwickler. Dabei sammelte er in diversen Projekten Erfahrungen und wurde immer wieder mit Problemen der Qualitätssicherung und des Testings konfrontiert. In der Freizeit absolviert er Bahnen im Schwimmbecken oder läuft durch die Wälder Schaffhausens.

Domenico Bernardo

Domenico Bernardo arbeitet seit 2001 in der AdNovum als Release Engineer, seit diesem Jahr als Head of Release Engineering. Wenn er nicht gerade mit dem Bau und der zeitgerechten Lieferung von Software Releases beschäftigt ist, geniesst er in voller Fahrt die Errungenschaften des italienischen Auto- und Motorrad-Designs.

Identity Management

Vor kurzem hat AdNovum für einen schweizerischen Finanzdienstleister ein IDM-System gebaut. Anhand der dabei gewonnenen Erfahrungen lassen sich beispielhaft die Herausforderungen und Strategien bei der Realisierung und Einführung eines solchen Systems aufzeigen.

Von Andreas Petralia

In einer zunehmend verteilten IT-Welt sehen sich Unternehmen damit konfrontiert, ihre Daten und Informationssysteme vor unerlaubten externen und internen Zugriffen schützen zu müssen. Dass interne wie externe Mitarbeiter für den Zugriff auf diese Ressourcen über eine Vielfalt von Identitäten verfügen, erschwert den Schutz der Daten und Systeme und ist für die Mitarbeiter selbst kompliziert und aufwändig. Aus diesem Grund gewinnt die Verwaltung von Identitäten als Thema zunehmend an Bedeutung.

Vielfältige Erwartungen

In einfachster Form steht Identity Management (IDM) für die zentrale Verwaltung von Identitäten, Passwörtern und Zugriffsrechten digitaler Nutzer. Als Buzzword weckt IDM jedoch vielfältige Erwartungen; IDM wird als Antwort gesehen auf Anforderungen bezüglich

- Produktivität (automatisches Erstellen von Accounts, Single Signon, Benutzerfreundlichkeit)
- Sicherheit (einheitliche Authentisierung, zuverlässige Löschung von nicht gebrauchten Accounts und Rechten, bessere Passwort-Policies)
- Compliance (Datenschutz, Archivierung, Auskunftspflicht)
- Kostendruck im IT-Management (Konsolidierung der ID Stores, Automatisierung der Administration, Zeitersparnis für Angestellte/Helpdesk)
- Flexibilität (schnelle und flexible IT-Welt mit ständig wechselnden Geschäftspartnern)

Eine solche Fülle von Anforderungen lässt sich kaum durch eine einzelne Applikation abdecken; in der Regel muss dazu ein verteiltes IDM-System aufgebaut werden.

Erfolg versprechendes Vorgehen

Wie lässt sich ein derart komplexes Vorhaben wie die Realisierung und Einführung eines IDM-Systems in die Praxis umsetzen? Werfen wir einen Blick auf das IDM-Projekt, das AdNovum vor kurzem für einen grossen Finanzdienstleister realisieren durfte. In einer Spezifikations- und Realisierungsphase von nur neun Monaten wurde ein IDM-System gebaut, das Business-Applikationen für Beratungsleistungen und Compliance/Archivierung sowie mehr als 2000 Benutzer bedient.

Der Projekterfolg basiert unter anderem darauf, dass bereits bei der Konzeption die relevanten Punkte prioritär behandelt wurden:

Von Anfang an wurde die Integration von bestehenden Systemen und Sicherheitskonzepten als Projektbestandteil erkannt und definiert. Bei der Bestandesaufnahme wurde jede Applikation hinsichtlich ihrer Integration ins IDM-System betrachtet. Von Interesse waren dabei die eingesetzte Technologie, Informationen zu den Benutzern (interne/externe) und deren Verwaltung sowie das angewandte Authentisierungs- beziehungsweise Autorisierungsmodell.

Ebenfalls zur Sprache kamen organisatorische Prozesse: Ablauf bei einer Berechtigungsmutation, Prozess bei Austritt eines Mitarbeiters, Prozess bei vergessenem Passwort etc. Nach der Bestandesaufnahme wurden die vorhandenen Probleme und Schwachstellen analysiert.

Der so gewonnene Überblick ergab bereits in der Konzeptphase erste Hinweise auf die zentralen Themen bei der Realisierung.

Ein IDM-System bedingt die Definition entsprechender technischer, aber vor allem auch betriebsorganisatorischer Prozesse. Die Einführung dieser Prozesse ist mit einem beachtlichen Aufwand verbunden. Während der Spezifikationsphase sollte man deshalb wo immer möglich versuchen, Prozesse und Abläufe zu vereinfachen. Klare Prozesse werden besser verstanden und können besser «gelebt» werden, was wiederum der Sicherheit zugute kommt.

Durch Vereinfachungen können zudem Kosten gespart und so der ROI beschleunigt werden: In der realisierten IDM-Applikation wurde von Anfang an darauf geachtet, dass die Benutzer Aufgaben möglichst selbständig erledigen können. So kann zum Beispiel jeder Benutzer seine persönlichen Daten selbst verwalten.

Als erfolgsentscheidend erwiesen sich zudem folgende Punkte:

- Die verschiedenen Applikationsverantwortlichen dürfen in einer IDM-Lösung nicht nur einen Mehraufwand sehen, das heisst, für die Applikationen muss ein Mehrwert in Aussicht gestellt werden (z.B. Auslagerung und Vereinfachung von Prozessen wie Authentisierung und Autorisierung, höhere Transparenz und Nachvollziehbarkeit, bessere Kontrolle und Überwachung).
- Bei einem IDM-Projekt sind viele verschiedene Organisationseinheiten und Personen betroffen und involviert, deshalb sollte ein solches Projekt mit der gebührenden Priorität behandelt und vom obersten Management getragen werden.

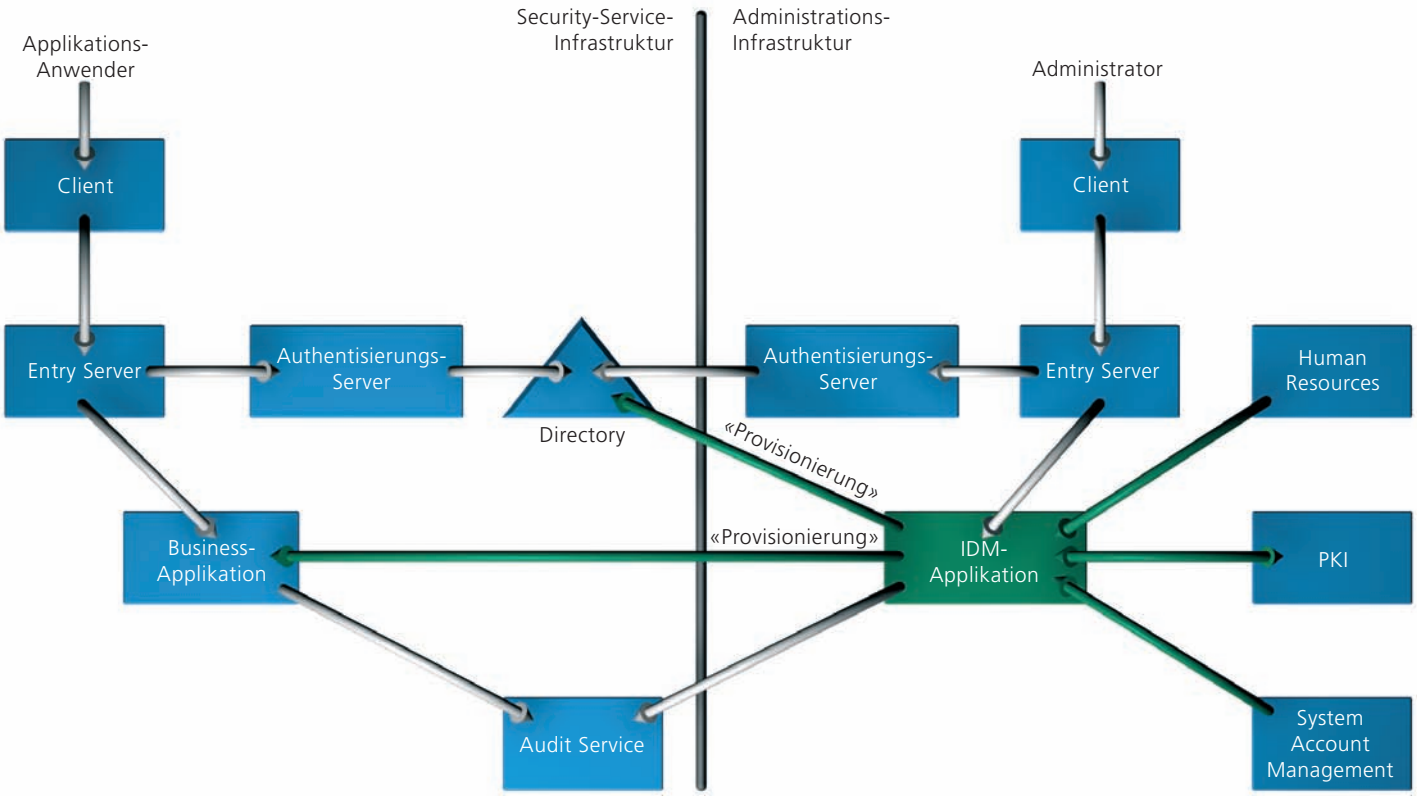
> Die Definition von technischen und vor allem von betriebsorganisatorischen Prozessen ist für ein IDM-System unumgänglich.

Modular aufgebaute Architektur

In Anbetracht des grossen Funktionsumfangs sollte ein IDM-System von Anfang an als verteiltes System mit klaren Kontrakten zwischen den Komponenten konzipiert werden. Ein System, das aus unabhängigen Komponenten aufgebaut ist, erlaubt neben einem besseren Verständnis der Problematiken auch ein schrittweises und priorisiertes Vorgehen bei der Spezifikation und Realisierung.

Besondere Beachtung verdient dabei die Sicherheit: Sie muss durch eine unternehmensweit einheitliche Sicherheitsarchitektur sowie durch Vorkehrungen im eigentlichen IDM-System als neuralgischem Angriffspunkt gewährleistet werden.

Die Kommunikation beispielsweise erfolgt im hier vorgestellten IDM-System authentisch, vertraulich und integer auf SSL. Alle dafür notwendigen Zertifikate werden durch das Zertifikats-Management-System (ZMS)



Architekturschema eines verteilten und modularen IDM-Systems.

verwaltet, das auf einer PKI-Applikation aufbaut.

Um eine Entkopplung zwischen Administrationskomponenten und hoch verfügbaren Runtime-Services zu erreichen, wurde das IDM-System in unserem Referenzprojekt in zwei Teile aufgeteilt:

> Ein modular aufgebautes IDM-System erlaubt ein schrittweises und priorisiertes Vorgehen bei Spezifikation und Realisierung.

• Administrationsinfrastruktur

In der Administrationsinfrastruktur findet die Datenverwaltung für das IDM statt. Den zentralen Teil bildet die IDM-Applikation, die für das Credential-, Account- und Permission-Management verantwortlich ist. Die Anbindung an das Human-Resources-System garantiert eine automatische Übernahme von Stammdaten und das Erkennen von Ein- und vor allem von Austritten von Mitarbeitern. Die Anbindung an zusätzliche Infrastrukturkomponenten (PKI, Audit Service, System Account Management etc.) erweitert den Funktionsumfang der IDM-Lösung.

Die in der Administrationsinfrastruktur bereitgestellten Daten werden dem Security Service zur Verfügung gestellt, wo sie für die Authentisierung und Autorisierung verwendet werden. In diesem Zusammenhang spricht man auch von «Provisionierung».

Die modulare IDM-Applikation besteht aus einer Core-Komponente, die keine Kenntnisse über die angeschlossenen Umsysteme besitzt. Der Core-Komponente liegt ein Datenmodell zugrunde, das Identitäten, Rollen, Berechtigungen und Credentials verknüpft. Über ein Mapping-Modul-Konzept kommuniziert die Core-Komponente mit Informationsquellen, von denen sie entsprechende aktuelle Daten erhält, und Empfängern, die sie mit den spezifisch relevanten Informationen versorgt.

• Security-Service-Infrastruktur

Die Security-Service-Infrastruktur (SSI) ist eine höchstverfügbare Sicherheits- und Authentisierungsinfrastruktur, die auf der Basis von Nevis Web realisiert wurde. Hier finden sich auch ein Audit Service und ein Verzeichnisdienst (Directory). Das Directory wird von der IDM-Applikation provisioniert und enthält somit alle im IDM-Modell modellierten und definierten Informationen. Zukünftigen Erweiterungen des Modells kann durch eine Erweiterung des LDAP-Schemas Rechnung getragen werden.

Zu den Nevis-Web-Komponenten zählen der Entry Server (ein sicherer Reverse Proxy) und der Authentisierungsdienst.

Der Entry Server (nevisProxy) dient als zentraler Einstiegspunkt für alle integrierten Applikationen (Business Application Backbone). Er stellt in Kombination mit dem Authentisierungsdienst (nevisAuth) sicher, dass nur authentisierte und autorisierte Benutzer auf die Applikationen zugreifen können.

Die Authentisierung kann mittels X.509-Zertifikaten, Challenge Response und weiteren Verfahren erfolgen. Nach einer erfolgreichen Authentisierung werden Benutzerattribute wie Name, Applikationsrollen und Authentisierungsstärke in ein Security Token geschrieben, das der Entry Server der Business-Applikation bei jeder Abfrage mitsendet.

Integration der Applikationen

Für die Einbindung einer Applikation ins IDM-System werden zwei Möglichkeiten unterstützt:

Für Neuentwicklungen und Nicht-Legacy-Applikationen wurde die Anbindung an die Nevis-Web-Infrastruktur vorgesehen. Bei dieser Einbindungsart genügt zum Provisionieren des Directory auf Seite der IDM-Applikation ein

generisches LDAP-Mapping-Modul. Für viele der anzubindenden Applikationen ist dies die effizienteste Einbindungsart. An die Applikation werden die folgenden Anforderungen gestellt: Proxy-fähiges Protokoll zwischen Client und Server, Auslagerung der Authentisierung und Einbindung des Security Token. Weiter kann eine Anbindung an bestehende Audit- und Autorisierungs-Services ins Auge gefasst werden.

Bei älteren und bereits vorhandenen Applikationen, bei denen eine Migration auf die Nevis-Infrastruktur nicht möglich bzw. zu teuer ist, kann entweder eine direkte Anbindung der Applikation an das Directory oder eine Anbindung direkt an die IDM-Applikation in Erwägung gezogen werden. Falls die IDM-Applikation die Provisionierung solcher Applikationen selbständig übernehmen soll, sind dafür jedoch eigene Mapping-Module mit entsprechenden applikationsspezifischen Schnittstellen erforderlich.

Offen für die Zukunft

Die Einführung eines IDM-Systems stellt einen vor allem in organisatorischer Hinsicht nicht zu unterschätzenden Aufwand dar, ist jedoch ein lohnendes Unterfangen. Werden die integrativen und organisatorischen Aspekte frühzeitig und prioritär behandelt, so steht dem Projekterfolg nichts im Wege.

Durch den modularen Aufbau seiner Komponenten ist das von der AdNovum entwickelte IDM-System auf die Anpassung an zukünftige Anforderungen bestens vorbereitet, und Erweiterungen sind auf einfache Weise realisierbar. An entsprechenden Wünschen und Ideen besteht denn auch kein Mangel, zum Beispiel:

- Föderation von Authentisierungsdomänen
- Unterstützung von Pseudonymen für den Datenschutz
- komplexere Berechtigungsvergabe (zum Beispiel regelbasiert)

Ideen im Bereich IDM gehen sogar so weit, dass man zusätzlich zu den Benutzeridentitäten auch die Identitäten von Rechnern, Applikationen und Services verwaltet. IDM entwickelt sich damit in Richtung eines umfassenden Informationssystems, welches die gesamte produktive Systemlandschaft abbildet.

Andreas Petralia

Andreas Petralia studierte an der ETH Zürich Elektroingenieur und arbeitete anschliessend in einer Start-up Company im Bereich der biometrischen Authentisierung.
Nach dem Wechsel zu AdNovum vor vier Jahren machte er anfänglich einen Abstecher in die Middleware-Entwicklung und festigte anschliessend sein IT-Security-Know-how als Entwickler bei Security-Projekten.
Als Projektleiter im hier vorgestellten Projekt wurde er mit den technischen und organisatorischen Aspekten bei der Herstellung und Einführung von Software vertraut.
Sein privates Identity Management befasst sich eher mit nicht technischen Dingen: sich zu Marathonläufen überreden lassen oder beim Squashen Energie ablassen.

Mandantenfähigkeit

Schlagwort oder Realität? Mandantenfähigkeit wird sehr unterschiedlich definiert und umgesetzt. Mit Finnova erhält eine Bank die Plattform, welche auf technischer, applikatorischer sowie Prozess-Ebene die Möglichkeit der Mandatenfähigkeit bietet.

Von Christian M. Winzenried, finnova AG Bankware

Finnova wurde auf der Basis modernster Technologien entwickelt. Die 3-Tier-Client/Server-Architektur ermöglicht dank dem Open-Interface-Konzept grosse Flexibilität bezüglich Erweiterbarkeit, womit auf zukünftige Markt- und Kundenbedürfnisse eingegangen werden kann.

Drei Sichten – ein gemeinsamer Nenner

Wann ist aus der Sicht einer Bank eine Standardlösung wie Finnova «mandantenfähig»? Welche Faktoren spielen für ein Rechenzentrum oder für das Application Management eine tragende Rolle?

Aus Sicht der Bank ist eine Plattform mandantenfähig, wenn

- die Prozessapplikationen mehrerer Banken auf einer Plattform betrieben werden können,
- Kompetenzzentren geführt werden können, sodass eine Bank für mehrere Banken Bankprozesse abwickelt (zum Beispiel Scanning Einzahlungsscheine), bei welchen sich die Bankgeschäfte weder für die Bank noch für den Bankkunden gegenseitig beeinflussen,
- die Stammdatenpflege (beispielsweise Valorendaten) von einer Bank für mehrere Banken geführt werden kann.

Aus Sicht eines Rechenzentrums ist eine Plattform mandantenfähig, wenn

- mehrere Mandanten auf einem Server betrieben werden können,
- mehrere Mandanten in einer einzelnen Datenbank-Instanz gespeichert werden können,
- das Monitoring für mehrere Mandanten auf einer Konsole erfolgen kann,
- die externen Systeme ebenfalls mandantenfähig sind,
- die mandantenfähigen Daten und Prozesse gegeneinander geschützt sind.

Kurzum: Die kostengünstigste Lösung aus Sicht eines Rechenzentrums wird durch die Minimierung der notwendigen Hardware, der systemnahen Software und der zu überwachenden Prozesse erreicht.

Der gemeinsame Nenner aus Sicht von Bank, Rechenzentrum und Application Management: Kostenreduktion.

Aus Sicht des Application Management hingegen ist eine Plattform dann mandantenfähig, wenn

- alle Daten sich im selben Oracle-Schema (Datenmodell) befinden,
- die Software und die Patches nur einmal installiert werden müssen,
- die Basis-Parametrisierung zentral erfolgen kann und nur einmal vorgenommen werden muss,
- die bankindividuelle Parametrisierung lediglich durch Delta-Parametrisierung zur Basis erfolgen kann,
- das Testing von Applikationen mit Standardparametrisierung zentral erfolgen kann.

Fazit: Die kostengünstigste Lösung aus Sicht des Application Management wird durch zentrale Daten- und Software-Haltung erreicht.

Die Mandantenfähigkeit optimal unterstützend, hilft das freie Schlüsselsystem im Finnova, die extern bekannten Nummern für Kunden, Konto, Portfolio, Kontoarten usw. weiterhin zu verwenden (also keine Transformation von alter in neue Nummer nötig!).

Heutzutage spielt auch immer mehr der 24-Stunden-Betrieb eine wichtige Rolle; Finnova ist gebaut für diese Hochverfügbarkeit.

Ob dabei alle historisierten Daten online verfügbar sein sollen, ist durch die Bank selbst parametrierbar.

finnova AG Bankware

finnova AG Bankware mit Sitz in Lenzburg (www.finnova.ch) ist einer der Marktplayer im Schweizer Markt für Gesamtbankenlösungen.
Die Finnova AG ist ein rein schweizerisches Softwarehaus und kann dabei auf mehr als dreissig Jahre Erfahrung in der Entwicklung professioneller IT-Lösungen für Finanzdienstleister aufbauen. Kombiniert mit profundem Branchen-Know-how macht dies Finnova zu einer ebenso modernen wie kostengünstigen Bankenplattform.
Die Fokussierung ist klar gerichtet auf die Weiterentwicklung der Gesamtbankenlösung. Dabei kann Finnova auf eine langjährige und erfolgreiche Entwicklungs- und Zusammenarbeitskultur mit Universalbanken und Privatbanken zählen.
Infolge der Expansion der Kundschaft werden Mitte dieses Jahres zwei Aussenstellen aufgebaut. Neben der Zentrale in Lenzburg wird der Bereich Application Management in Seewen SZ domiziliert sein, und in Chur werden weitere Entwicklerteams arbeiten.

Impressum

Herausgeber:
AdNovum Informatik AG
Corporate Marketing
Röntgenstrasse 22
CH-8005 Zürich
Telefon 044 272 61 11
Telefax 044 272 63 12
E-Mail info@adnovum.ch
www.adnovum.ch
Verantwortlich und Redaktion:
Manuel Ott
Gestaltung und Realisation:
Rüegg Werbung, Zürich
Fotografie:
Gerry Nitsch, Zürich