

IBM Produkte für Portale und Onlinezusammenarbeit als Basis einer SOA

*Wie sich durch Einsatz von IBM Produkten
für Portale und Onlinezusammenarbeit ein
sofortiger geschäftlicher Nutzen erzielen lässt
– eine technische Anleitung*

*Dr. Barry Devlin
Dublin Software Laboratory
IBM Software Group*

Contents

- 2 Einleitung**
- 3 Hintergrund**
- 5 Kundenbetreuung ohne SOA**
- 11 Kundenbetreuung nach SOA-Art**
- 17 Implementierung einer SOA mit
IBM Produkten für Portale und
Onlinezusammenarbeit**
- 25 Zusammenfassung**

Einleitung

Dieses White Paper erläutert, wie IBM Produkte für Portale und Onlinezusammenarbeit für den raschen Aufbau einer ersten serviceorientierten Architektur (SOA) eingesetzt werden können. Dabei werden traditionelle Anwendungen integriert und aktualisiert und die Benutzerproduktivität durch eine moderne, portalbasierte Schnittstelle deutlich verbessert. Wir wollen dies anhand eines vereinfachten, aber realistischen Szenarios aus dem Bereich der Kundenbetreuung erörtern.

Wir beginnen mit einer kurzen Beschreibung der Begriffe und Faktoren im Zusammenhang mit einer SOA.

Anschließend werden wir unser Szenario vorstellen: Am Beispiel eines fiktiven Unternehmens werden wir eine für die Kundenbetreuung typische Umgebung beschreiben. Wir betrachten diese Umgebung in erster Linie aus der Sicht einer Mitarbeiterin im Kundenservice, die jeden Tag mit dem System für die Kundenbetreuung arbeiten musste. Wir sehen uns jedoch auch die Entwicklung und Struktur des Systems aus der Perspektive eines Architekten an. Auf diese Weise erkennen wir sowohl die Probleme, die die Servicemitarbeiter mit dem System hatten, als auch die Schwierigkeiten, mit denen die IT-Abteilung bei der Bereitstellung dieses Services konfrontiert war.

Die zwei wichtigsten Kapitel dieses White Papers beschreiben die SOA-basierte Lösung. Sie skizzieren die neue Architektur und die technischen Aspekte, die bei der Integration vorhandener Anwendungen in die SOA berücksichtigt werden mussten. Wir untersuchen kurz, wie vorhandene oder traditionelle Anwendungen in wieder verwendbare Services aufgeteilt werden können (und müssen) und wie diese Services in modularen Anwendungen kombiniert werden können. Wir stellen die ausgewählten Middlewareprodukte – IBM WebSphere Portal, IBM WebSphere Portlet Factory und IBM Lotus Sametime – sowie einige Lösungen und die Gründe für die Auswahl dieser Produkte vor.

Schließlich werden wir die Vorteile der SOA-Lösung, sowohl für die Benutzer als auch die IT-Mitarbeiter, aufzeigen. Die Benutzer profitieren von höherer Produktivität, geringerem Schulungsaufwand und der Chance auf mehr Innovation in ihrer täglichen Arbeit. Die IT-Abteilung wird entlastet, da sie sich statt auf Routinewartungsarbeiten an vorhandenen Anwendungen auf die schnelle und effektive Bereitstellung neuer Funktionen konzentrieren kann.

Highlights

Eine SOA unterstützt die Serviceorientierung auf der Basis offener Standards. Unter „Service“ versteht man in diesem Zusammenhang eine klar abgegrenzte, definierte und wiederholt durchführbare Geschäftsaufgabe, die auf standardisierte Weise aufgerufen werden kann.

Eine SOA ermöglicht die Modellierung, Entwicklung, Zusammenstellung, Implementierung und das Management flexibler, integrierter Anwendungen, die aus wieder verwendbaren Services bestehen, die unabhängig von den Anwendungen und Plattformen sind, auf denen sie ausgeführt werden.

Modulare Anwendungen werden meist in einer Portalarchitektur implementiert, wobei einzelne Portlets im Allgemeinen eigenständigen Services entsprechen.

Hintergrund

Eine serviceorientierte Architektur ist, einfach ausgedrückt, eine IT-Architektur, die die Serviceorientierung auf der Basis offener Standards unterstützt. Unter „Service“ versteht man in diesem Zusammenhang eine klar abgegrenzte, definierte und wiederholt durchführbare Geschäftsaufgabe, die auf standardisierte Weise aufgerufen werden kann. Beachten Sie, dass es hierbei ausschließlich um geschäftliche Aspekte geht. Der Umfang dieser Aufgabe kann sehr eng, aber auch relativ weit gefasst sein. Es kann sich also um eine einfache Aufgabe handeln, die aus nur einem Schritt besteht, z. B. die Einrichtung der Rechnungsadresse eines Kunden, oder um eine komplexere Aufgabe, die mehrere Schritte und eine ganze Reihe möglicher Ergebnisse umfasst, z. B. die Eröffnung eines neuen Kontos. Die Services können zudem miteinander verknüpft sein, d. h. ein Service kann einen anderen aufrufen, der dann eine untergeordnete Aufgabe durchführt, oder sogar feststellen, ob eine andere Aufgabe aufgerufen wird oder nicht. Dank dieser Form der Serviceintegration und der Verknüpfung von Ergebnissen sind die Anwendungen sowohl flexibel als auch integriert – und erfüllen damit zwei der wichtigsten Voraussetzungen in der modernen Geschäftswelt, die die Basis für ein wirklich flexibles Unternehmen bilden.

Eine SOA ermöglicht so die Modellierung, Entwicklung, Zusammenstellung, Implementierung und das Management flexibler, integrierter Anwendungen, die aus wieder verwendbaren Services bestehen, die unabhängig von den Anwendungen und Plattformen sind, auf denen sie ausgeführt werden. Diese Prozesse werden durch einen Governance-Ansatz und bewährte Verfahren (Best Practices) auf der Basis praktischer Erfahrung unterstützt, wie in Abbildung 1 (siehe Seite 4) dargestellt. Solche flexiblen, integrierten Anwendungen bezeichnet man gemeinhin als modulare Anwendungen. Sie bilden die Basis für anpassungsfähige Geschäftsprozesse. Früher war ein Geschäftsprozess eine größtenteils statische Gruppe von Aufgaben innerhalb einer bestimmten Abteilung. Implementiert wurde er in einer einzelnen, in sich geschlossenen, traditionellen Anwendung. In Zukunft wird ein Geschäftsprozess auf modularen Anwendungen basieren, die aus flexibler Geschäftslogik zwischen Services (bekannt als Workflow) bestehen. Die unabhängigen und einfach modifizierbaren Services erstrecken sich dabei auf mehrere Abteilungen. Modulare Anwendungen werden meist in einer Portalarchitektur implementiert, wobei einzelne Portlets im Allgemeinen eigenständigen Services entsprechen.

Highlights

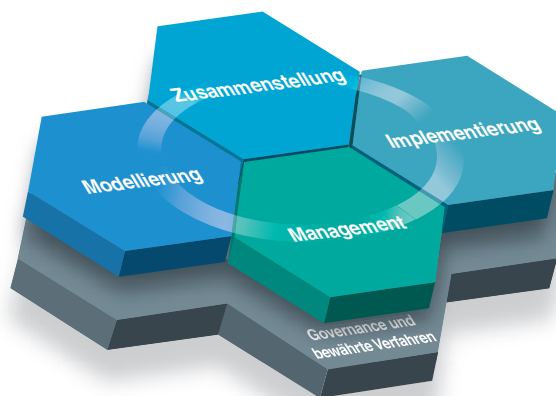


Abbildung 1: Serviceorientierte Architektur

In der SOA-Umgebung sind Portlets die wichtigsten Instrumente der Benutzerschnittstelle, und Aufgabenseiten stellen die grundlegenden Geschäftsfunktionen bereit.

Obwohl sich das Hauptaugenmerk meist auf Services richtete, die aus traditionellen, betrieblichen, geschäftsbereichsspezifischen Anwendungen extrahiert werden können, lassen sich auch andere Arten von Services auf ebenso einfache Weise in eine SOA einbinden. Zu den wichtigsten Services dieser Art gehören Services für Portale und Onlinezusammenarbeit. Sie bilden die Grundlage für die Benutzerinteraktion mit dem Workflow und sind von entscheidender Bedeutung für die heute geforderte deutliche Steigerung der Benutzerproduktivität. In dieser Umgebung sind Portlets die wichtigsten Instrumente der Benutzerschnittstelle, und Aufgabenseiten stellen die grundlegenden Geschäftsfunktionen bereit. Ein koordinierter Workflow ist im Grunde ein Geschäftsprozess, bei dem der Portalserver einem Benutzer Aufgabenseiten und Portlets auf der Basis der Rolle des Benutzers präsentiert. Ein solcher Workflow unterstützt außerdem die für die aktuelle Rolle und Situation des Benutzers am besten geeignete Navigation und Verbindung dieser Aufgaben und Portlets.

Highlights

Eine SOA erlaubt die Erstellung von Geschäftsprozessen, die die Arbeit im Team durch Möglichkeiten der Onlinezusammenarbeit, auch über große Entfernungen hinweg, unterstützen und fördern.

In diesem Kapitel stellen wir Ihnen eine typische traditionelle Anwendungsumgebung am Beispiel eines fiktiven Unternehmens vor.

Das Reklamationsbearbeitungssystem des Unternehmens wurde ursprünglich dafür entwickelt, jede Funktion bereitzustellen, die ein Mitarbeiter im Kundenservice benötigt.

In Verbindung mit Services für Portale und Onlinezusammenarbeit ermöglicht eine SOA die Erstellung von Geschäftsprozessen, die die Arbeit im Team durch Möglichkeiten der Onlinezusammenarbeit, auch über große Entfernungen hinweg, unterstützen und fördern. Solche Ansätze können zu echter Innovation im Unternehmen beitragen, indem sie den Zugang zum Wissen der Benutzer und zu den auf mehrere Datenrepositorys verteilten Informationen erleichtern.

Kundenbetreuung ohne SOA

In diesem Kapitel stellen wir Ihnen eine typische traditionelle Anwendungsumgebung am Beispiel eines fiktiven Unternehmens vor, das wir Galactic Resource Provisioning (GReP) nennen wollen. Dabei handelt es sich um einen landesweit tätigen Einzelhändler, der Tuning-Kits und Zubehör für Autos vertreibt. Das Unternehmen betreut seine Kunden über ein zentrales Call-Center, das Anfragen, Beschwerden und technische Fragen bearbeitet. GReP verlangte von den Call-Center-Mitarbeitern, im nächsten Jahr neue Umsätze mittels Cross-Selling zu generieren. Doch die Mitarbeiter leisteten Widerstand, weil sie ihrer Ansicht nach ohnehin schon überlastet waren.

Sehen wir uns einmal die Aufgaben an, die Maria, eine der Call-Center-Mitarbeiterinnen, während des Telefonats mit einem Kunden durchführen musste. Ein großes Fenster auf Marias Bildschirm gehörte zu einer traditionellen Client-Server-Anwendung mit der Bezeichnung CMS (Complaints Management System). Dieses Reklamationsbearbeitungssystem stand im Mittelpunkt fast aller Aufgaben, die sie erledigte. Ursprünglich war das CMS dafür entwickelt worden, jede Funktion bereitzustellen, die ein Mitarbeiter im Kundenservice benötigt. Tatsächlich hatte das System dieses Ziel in der Vergangenheit fast erreicht. Doch Veränderungen in Geschäftsumfeld und Technologie hatten die Rolle des CMS eingeschränkt.

Highlights

Aufgrund von Änderungen im Kundenverhalten und an den IT-Systemen des Unternehmens mussten die Servicemitarbeiter ständig zwischen der Kundendatenbank, dem CMS und dem SAP-System hin- und herwechseln, um die Kundennummer eines Anrufers abzugleichen und sein Einkaufsprofil und den Gewährleistungsstatus zu verifizieren.

Beim Eingang eines Anrufs versuchte das CMS, die Telefonnummer in der Kundendatenbank zu finden und Details zum jeweiligen Kunden anzuzeigen. Leider war diese Funktion nicht mehr so effektiv wie früher, weil die Kunden inzwischen zunehmend von Mobiltelefonen aus anriefen oder ihre Rufnummer nicht übermittelten. In vielen Fällen musste Maria den Anrufer nach seinem Namen fragen. Dann stattete sie dem SAP-System den ersten von mehreren Besuchen ab, um nach der Kundennummer zu suchen. Wenn sie diese gefunden hatte, gab sie sie in das CMS ein, das dann nach den übrigen Kundendaten suchte.

Danach musste Maria überprüfen, ob der Anrufer das Produkt tatsächlich bei GReP gekauft hatte und ob die Gewährleistung noch gültig oder schon abgelaufen war. Hierfür musste Maria dem SAP-System weitere Besuche abstatten und dann die Ergebnisse ihrer Suche kopieren und in das CMS einfügen. Daher war das SAP-Fenster auf Marias Bildschirm immer offen, und sie verwendete viel Zeit darauf, zwischen dem CMS und dem SAP-System hin- und herzuwechseln und Informationen aus dem einen System zu kopieren und ins andere einzufügen. Je nachdem, welche Produkte der Kunde besaß und auf welchem Weg er sie gekauft hatte, hätte Maria mehrere offene SAP-Fenster gebraucht, um den Anspruch des Anrufers auf Support zu überprüfen. Das kostete sie viel Zeit und Arbeit, die ihrer Meinung nach „reine Verschwendung“ waren.

Um fair gegenüber der IT-Abteilung zu sein, muss angemerkt werden, dass das CMS ursprünglich alle Datenübertragungen zwischen dem System selbst und der Kunden- und Produktdatenbank durchführte. Als GReP jedoch einen anderen Anbieter von Autozubehör aufkaufte, migrierte das Unternehmen seine selbst entwickelte Kundendatenbank auf das SAP-System, das das andere Unternehmen eingesetzt hatte. Die Integration von CMS und SAP-System hatte sich als zu schwierig erwiesen.

Highlights

Die Arbeitsoberfläche der Call-Center-Mitarbeiter enthielt unzählige offene Fenster. Sie alle gehörten zu Systemen, auf die die Mitarbeiter zugreifen mussten, um die Anfragen von Kunden zu beantworten.

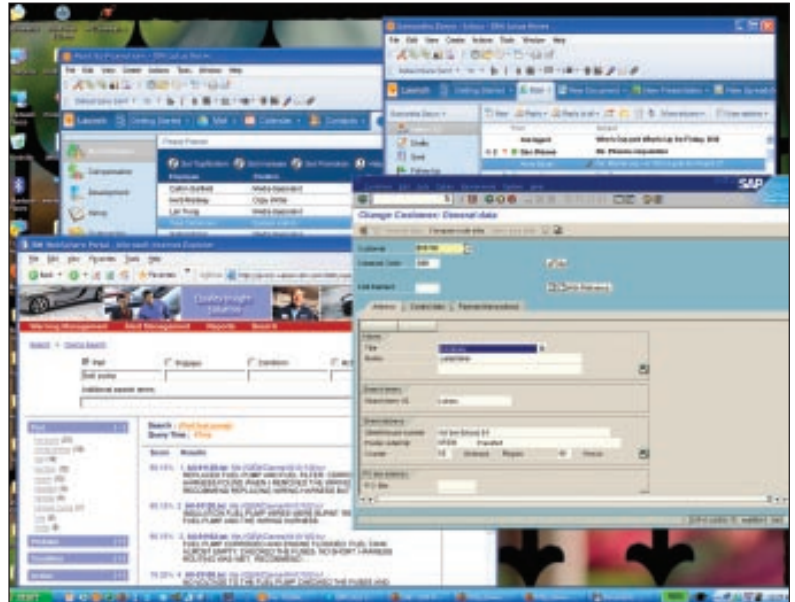


Abbildung 2. Screenshot eines traditionellen Call-Center-Systems

Endlich war Maria bereit, die eigentliche Anfrage des Kunden zu beantworten. Als sie Details der Anfrage oder Beschwerde in das CMS eingab, ermittelte das System automatisch bestimmte Schlüsselwörter, die in der Antwortdatenbank – einem Content-Management-System, das als weiteres Fenster auf Marias Bildschirm erschien – verwendet werden konnten. Die Schlüsselwortsuche ist zwar eine leistungsstarke und nützliche Funktion, doch eine nahtlose Integration zwischen CMS und Antwortdatenbank hatte sich in der Vergangenheit als unmöglich herausgestellt. Also musste Maria die Schlüsselwörter manuell kopieren und einfügen – und eine weitere Schnittstelle verwenden, mit der sie sich auskennen musste.

Highlights

Wenn Maria die Anfrage des Kunden mithilfe der Antwortdatenbank beantworten konnte, erstellte sie einen Querverweis im CMS und schloss den Reklamationseintrag. Wenn nicht, kontaktierte sie das Second-Level-Support-Team. Dafür benötigte sie eine weitere Anwendung – mittlerweile die vierte, die auf ihrem Bildschirm erschien: IBM Lotus Notes, die Standard-E-Mail-Anwendung des Unternehmens. Maria kopierte Details zu dem Problem aus dem CMS in eine E-Mail an Second-Level-Support und vereinbarte mit dem Kunden, ihn zurückzurufen. Sie musste später selbst daran denken, die Antwort vom Second-Level-Support rechtzeitig an den richtigen Kunden zu übermitteln, entweder telefonisch oder per E-Mail.

Dieses Szenario macht deutlich, dass die Arbeit im Kundenservice eine komplizierte Angelegenheit war:

Die Arbeit im Kundenservice war eine komplizierte Angelegenheit aufgrund zahlreicher Faktoren: einer chaotischen Arbeitsoberfläche, unterschiedlichen Benutzerschnittstellen und Integrationsstufen, der Notwendigkeit, Informationen manuell zwischen Systemen zu kopieren und einzufügen, und fehlender Workflow-Unterstützung. Das Ergebnis war, dass die Mitarbeiter nicht so produktiv waren, wie sie eigentlich hätten sein können.

- *Aufgrund einer chaotischen und überladenen Arbeitsoberfläche war es schwierig, den Überblick über die zu erledigenden Aufgaben zu behalten.*
- *Die Verwendung unterschiedlicher Benutzerschnittstellen bedeutete höhere Schulungskosten und verlangsamte die Benutzeraktionen.*
- *Das manuelle Kopieren und Einfügen von Daten zwischen unterschiedlichen Anwendungsfenstern war zeitaufwendig und fehleranfällig.*
- *Unterschiedliche Ebenen der Integration zwischen Anwendungen stifteten Verwirrung.*
- *Da es an effektiver Workflow-Unterstützung fehlte, mussten die Call-Center-Mitarbeiter Anfragen, die nicht sofort beantwortet werden konnten, selbst nachverfolgen.*

Das Ergebnis: Die Mitarbeiter im Kundenservice waren bei weitem nicht so produktiv, wie sie eigentlich hätten sein können, und mussten Ad-hoc-Prozesse erfinden, um ihre Arbeit zu bewältigen. Als das Unternehmen die Rolle der Servicemitarbeiter auf Cross-Selling und Umsatzgenerierung ausweiten wollte, stieß es daher auf erhebliche Widerstände und Schwierigkeiten.

Highlights

Das CMS war ursprünglich dafür entwickelt worden, alle Anwendungen, die die Mitarbeiter im Kundenservice brauchten, vollständig zu integrieren.

Infolge des geschäftlichen und technischen Wandels ging ein Großteil der Integration des CMS verloren. An die Stelle des automatischen Informationsaustauschs trat die manuelle Übertragung von Daten durch die CMS-Benutzer.

Die traditionelle Systemarchitektur

Wie bereits erwähnt, war das Reklamationsbearbeitungssystem CMS des Unternehmens ursprünglich als vollständig integrierte Anwendung geplant worden, die alle Anforderungen der Mitarbeiter im Kundenservice erfüllen sollte. Diese Mitarbeiter benötigen für ihre Arbeit Zugriff auf verschiedene Datensätze, die von anderen Abteilungen des Unternehmens erstellt und gepflegt werden, z. B. Produktdaten, Informationen zu Problemlösungen, Kundenkontaktinformationen und Bestell-/Kaufdaten. Außerdem erstellen sie auch selbst Informationen, z. B. Problemlösungen und Kundenkontaktprotokolle. Das CMS war so konzipiert worden, dass jede Nacht Kopien von sich langsam ändernden Informationen, z. B. Produkt- und Problemlösungsdaten, empfangen und auf einem eigenen Server gespeichert wurden. Der Zugriff auf Kunden- und Bestell-/Kaufdaten sollte direkt über eine spezielle Anwendungsprogrammierschnittstelle (API) zum Auftragseingabesystem erfolgen. Selbst E-Mail war vollständig in das ursprüngliche Konzept des CMS integriert.

Doch infolge des geschäftlichen und technischen Wandels ging ein Großteil der Integration des CMS verloren. Nach Einführung der Lotus Notes-Software wurde die integrierte E-Mail-Schnittstelle des CMS nicht mehr genutzt, da die Benutzer stattdessen die mit vollem Funktionsumfang ausgestattete Lotus Notes-Umgebung verwendeten. Nach der Umstellung auf Standardsoftware – ein SAP-System für die Kundenbetreuung und IBM WebSphere Content Discovery Server¹ für den Zugriff auf die Antwortdokumentation –, wurden die Migration und Wartung der Dateitransfer- und API-Schnittstellen immer schwieriger. An die Stelle des automatischen Austauschs von Informationen trat die manuelle Übertragung von Daten durch die CMS-Benutzer, und der Einsatz der mit allen Funktionen ausgestatteten SAP- und WebSphere Content Discovery Server-Benutzerschnittstellen wurde zur Norm. Dies wird in Abbildung 3 (siehe Seite 10) deutlich. Darin ist die Systemarchitektur unmittelbar vor der Entscheidung für die Implementierung einer SOA dargestellt.

Auch wenn das hier beschriebene Ausmaß an Desintegration ein wenig übertrieben ist, haben doch die allermeisten Unternehmen, die integrierte Anwendungen in irgendeinem Geschäftsbereich entwickelt haben, ähnliche Probleme erlebt. Die Gründe lassen sich grob in zwei Kategorien einteilen: geschäftlicher Wandel und technischer Fortschritt.

Highlights

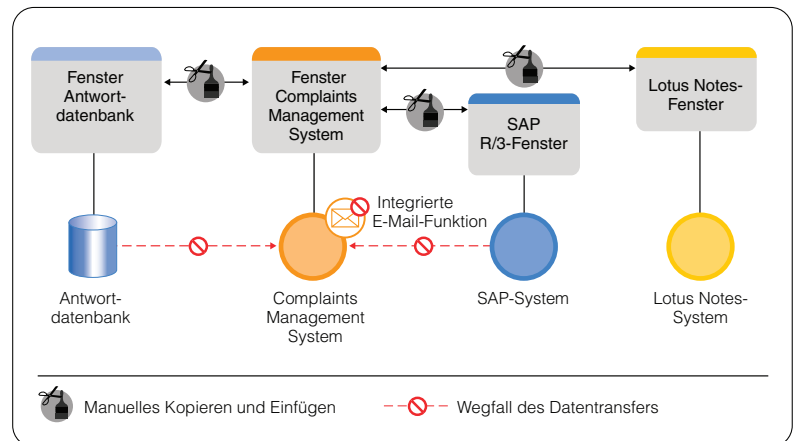


Abbildung 3. Traditionelle Architektur des Complaints Management System

Im Folgenden sind verschiedene geschäftliche Veränderungen aufgeführt, die integrierte Anwendungen unterbrechen:

Geschäftliche Veränderungen, die zu Unterbrechungen führen können, sind beispielsweise Fusionen und Übernahmen, neue Geschäftsprozesse, die infolge von Marktanforderungen entstanden sind, Outsourcing und Partnerschaften, die neue Schnittstellen erfordern, und das schnelle Tempo im On Demand Business.

- Fusionen und Übernahmen verursachen Unterbrechungen bei vorhandenen Anwendungen und Schnittstellen – häufig kann die IT-Abteilung nicht schnell genug reagieren.
- Änderungen in Geschäftsprozessen infolge von Marktanforderungen führen dazu, dass neue Integrationspunkte erforderlich sind und alte nicht mehr gebraucht werden.
- Outsourcing und Partnerschaften haben zur Folge, dass Funktionen in integrierten Anwendungen durch Schnittstellen zu externen IT-Systemen ersetzt werden.
- Die Anforderungen im On Demand Business, das von raschem Wandel geprägt ist, lassen sich mit den zwangsläufig langsamen Wartungszyklen einer integrierten Anwendung nur schwer erfüllen.

Highlights

Technische Veränderungen, die Unterbrechungen verursachen können, sind beispielsweise die Einführung von Standardsoftwareanwendungen, die Schnittstellen zu integrierten Anwendungen unterbrechen, sowie neue Hardwareplattformen, Betriebssysteme, Datenbanken und Middleware oder Paradigmenwechsel wie die Umstellung von der zentralisierten Datenverarbeitung auf Client-Server-Systeme.

Die traditionelle Herangehensweise an eine desintegrierte Umgebung kann einen erneuten Verlust der Integration des Systems nicht verhindern, weil die ursprünglichen fest „verdrahteten“ Schnittstellen durch neue, ebenfalls fest „verdrahtete“ Schnittstellen ersetzt werden.

Der technische Fortschritt wirkt sich ähnlich auf integrierte Anwendungen aus:

- *Die Implementierung von Standardsoftware unterbricht Schnittstellen zu einer integrierten Anwendung und kann sie im Extremfall ganz oder teilweise ersetzen.*
- *Änderungen bei Hardwareplattformen, Betriebssystemen, Datenbanken oder Middleware können erhebliche Unterbrechungen von integrierten Anwendungen verursachen.*
- *Paradigmenwechsel, wie z. B. die Umstellung von der zentralisierten Datenverarbeitung auf eine Client-Server-Umgebung, können das grundlegende Entwicklungsziel einer integrierten Anwendung verändern.*

Infolge dieser Faktoren hatte das CMS, genau wie viele andere integrierte Anwendungen in der Branche, seine Funktionsfähigkeit größtenteils verloren. Eine einst in höchstem Maße integrierte Systemarchitektur hatte sich in ihre Einzelteile aufgelöst und wurde nur noch von den Endbenutzern selbst zusammengehalten. Dieser Zustand war natürlich nicht hinnehmbar, und genau hier konnte eine SOA Abhilfe schaffen.

Kundenbetreuung nach SOA-Art

Bei der traditionellen Herangehensweise an eine desintegrierte Umgebung wie die oben beschriebene wird zunächst die Architektur des Systems untersucht. Dabei wird festgestellt, dass das Hauptproblem im Fehlen von automatisierten Schnittstellen zwischen den Systemen liegt. Die IT-Abteilung beginnt dann mit der Entwicklung und Erstellung neuer Batch- und Echtzeitschnittstellen. Damit lässt sich zwar möglicherweise ein Großteil des Problems lösen, doch tatsächlich gelangt man nur zurück zum Status des Systems, bevor es sich aufzulösen begann. Einen erneuten Verlust der Integration des Systems kann man jedoch nicht verhindern, da die ursprünglichen fest „verdrahteten“ Schnittstellen durch neue, ebenfalls fest „verdrahtete“ Schnittstellen ersetzt werden.

Eine serviceorientierte Architektur geht anders an das Problem heran. Sie zwingt die IT-Abteilung dazu, die Situation aus geschäftlicher und nicht nur aus technischer Perspektive zu sehen. Das bedeutet nicht, dass der technische Standpunkt unwichtig ist. Denn die moderne Technologie bietet viele Möglichkeiten, das Unternehmen auf neue Weise zu unterstützen, wie wir später noch sehen werden. Wenn die IT-Abteilung geschäfts- und serviceorientiert vorgeht, kann sie die zentralen Fragen stellen: Was wollen die Benutzer eigentlich erreichen? Und wie können wir ihnen helfen, ihre Ziele produktiver zu erreichen?

Highlights

Wenn man herausfinden möchte, was die Benutzer aus geschäftlicher Sicht erreichen wollen, muss man zum einen die entsprechenden Services bestimmen und zum zweiten die Services, die aus Sicht der Benutzer eng miteinander verknüpft sind, in Gruppen zusammenfassen.

In unserem Szenario haben wir zwei Benutzerrollen definiert, die von sechs Services unterstützt werden.

Geschäftsservices und Benutzerrollen

Die Beantwortung dieser Frage – was wollen die Benutzer erreichen? – ist ein wiederholt ablaufender Prozess, der zwei wichtige Bereiche umfasst:

- *Ermittlung der Services, d. h. der klar abgegrenzten und wiederholt durchführbaren Geschäftsaufgaben, die von den Benutzern ausgeführt werden*
- *Zusammenfassung der Services, die aus Sicht der Benutzer eng miteinander verknüpft sind, in Rollen, die die Benutzer zu unterschiedlichen Tageszeiten einnehmen*

Die Ermittlung von Services und Rollen kann auf unterschiedlichen Ebenen in einem Unternehmen ablaufen. Sie kann auf Unternehmensebene durchgeführt werden, um das gesamte Unternehmen umzugestalten, oder auf Abteilungsebene, um das Anwendungsportfolio und die Erfordernisse eines bestimmten Unternehmensbereichs neu zu bewerten. Solche Analysen sind nützlich, können sich aber als recht zeitaufwendig herausstellen. In unserem Szenario haben wir uns jedoch auf den relativ eng gefassten Bereich der Kundenbetreuung konzentriert. Daher konnten wir davon ausgehen, dass die Analyse schneller und einfacher vonstatten gehen und zur Ermittlung einer Reihe unterschiedlicher Services und Benutzerrollen führen würde. In unserem Beispielszenario haben wir zwei Benutzerrollen definiert, die von sechs Services unterstützt werden.

Die primäre Rolle eines Mitarbeiters im Kundenservice ist das Kontaktmanagement. Bei dieser Rolle geht es größtenteils darum, zu reagieren statt zu agieren. Die Call-Center-Mitarbeiter bearbeiten eingehende Anfragen, identifizieren Kunden, prüfen deren Anspruch auf Support und beantworten einfache Fragen. Die zweite Rolle ist die Suche nach Lösungen. Dabei wirken die Mitarbeiter proaktiv an der Erstellung neuer Antworten mit, kommunizieren mit dem Second-Level-Support-Team und nehmen erneut Kontakt zum Kunden auf, um Lösungen mit ihm zu erörtern.

Highlights

Die tägliche Arbeit eines Mitarbeiters im Kundenservice spielt sich überwiegend in der Rolle „Kontaktmanagement“ ab. Es gibt zwei Gründe für den Wechsel in die zweite Rolle: Der Mitarbeiter findet keine Antwort auf die Frage des Kunden und muss das Problem an Second-Level-Support weiterleiten. Oder das Second-Level-Support-Team präsentiert eine neue Lösung, die der Servicemitarbeiter dem Kunden übermitteln muss.

Wenn die Servicemitarbeiter, wie von GRP beabsichtigt, künftig formell für Cross-Selling-Aktivitäten verantwortlich sein sollen, werden die Services im Zusammenhang mit dieser Funktion in einer dritten Rolle zusammengefasst.

Jede Rolle besteht aus einer Reihe von Services, von denen einige auf eine bestimmte Rolle beschränkt sind und andere in mehreren Rollen vorkommen. Die Services werden den Benutzern in der Regel als Portlets in einer portalbasierten Umgebung präsentiert. Die folgende Tabelle zeigt die Services und die zugehörigen Rollen an.

Jede Rolle besteht aus einer Reihe von Services, von denen einige auf eine bestimmte Rolle beschränkt sind und andere in mehreren Rollen vorkommen.

Service	Rolle
Bearbeitung von Anfragen	Kontaktmanagement, Suche nach Lösungen
Suche nach Antworten	Kontaktmanagement, Suche nach Lösungen
Suche nach Kunden	Kontaktmanagement
Produktanfrage	Kontaktmanagement
Onlinezusammenarbeit	Suche nach Lösungen
Aufgabenerinnerung	Kontaktmanagement, Suche nach Lösungen

Die Bearbeitung von Anfragen ist ein Service, der für beide Rollen von zentraler Bedeutung ist. Über diesen Service werden Anfragen von Kunden verfolgt und dokumentiert, akzeptable Antworten protokolliert und Anfragen abgeschlossen.

Die Suche nach Antworten ist ein Service, der auf der Basis von Schlüsselwörtern die Datenbank nach bekannten Lösungen durchsucht.

Highlights

Die Suche nach Kunden ist ein Service, der den Namen eines Kunden als Input akzeptiert. Wenn der Name nicht eindeutig ist, stellt der Service eine Liste in Frage kommender Kundennummern und Adressen bereit und ermöglicht eine weitere Eingrenzung der Suche. Wenn der Kunde bekannt ist oder eindeutig identifiziert wurde, präsentiert der Service „Produktanfrage“ eine Liste von Produkten im Besitz des Kunden und zeigt deren Gewährleistungsstatus an.

Der Service „Onlinezusammenarbeit“ bietet E-Mail-Funktionalität, Instant-Messaging-Unterstützung und Zugriff auf Kontaktlisten.

***Wir müssen die Frage beantworten:
Wie können wir den Benutzern helfen,
ihre Ziele produktiver zu erreichen?***

Der Service „Aufgabenerinnerung“ liefert dem Servicemitarbeiter Informationen über Maßnahmen, die er durchführen muss. Dieser Service wird meistens eingesetzt, um den Mitarbeiter darüber zu informieren, dass eine Antwort für ein zuvor an Second-Level-Support weitergeleitetes Problem gefunden wurde und der Kunde nochmals kontaktiert werden sollte.

Modulare Anwendungen, Vorlagen und Workflows

Die zweite Frage, die wir uns gestellt haben, lautet: Wie können wir den Benutzern helfen, ihre Ziele auf produktivere Weise zu erreichen? Auf Architekturebene erreichen wir dies, indem wir vorlagenbasierte, modulare Anwendungen erstellen und Workflows innerhalb dieser Anwendungen ermöglichen.

***Auf Architekturebene erreichen wir dies,
indem wir vorlagenbasierte, modulare
Anwendungen erstellen und Workflows
innerhalb dieser Anwendungen
ermöglichen.***

Eine modulare Anwendung ist, einfach ausgedrückt, eine Anwendung, die eine Gruppe vorhandener Komponenten in einer einzelnen, zusammenhängenden Einheit für Endbenutzer zusammenfasst. Da sie auf der Wahrnehmung der Benutzer basiert, sind die Grenzen dieser Einheit möglicherweise etwas willkürlich. In unserem Szenario bilden alle Funktionen, die ein Mitarbeiter im direkten Kundenkontakt und in der Problemlösung ausführt, eine einzelne modulare Anwendung. Wir haben dies auf der Basis der Tatsache festgelegt, dass die einzelnen Komponenten und deren Interaktionen eine Gruppe zusammenhängender Geschäftslogik für den Benutzer implementieren.

In einer SOA findet sich Geschäftslogik auf zwei Ebenen: in den Services selbst und in der Verbindung zwischen ihnen. Die Geschäftslogik innerhalb von Services ist grundlegender und weniger veränderlich als die zwischen den Services. Der Service „Suche nach Kunden“ umfasst beispielsweise die standardisierte und erforderliche Geschäftslogik, die eine eindeutige ID für jeden Kunden fordert. Ein Service, der Geschäftsinformationen ändert, beinhaltet häufig sehr umfangreiche Geschäftslogik. In unserem Beispiel enthält der Service „Bearbeitung von Anfragen“ die Logik für das Erstellen neuer Verweise auf Anfragen, das Extrahieren von Schlüsselwörtern aus Reklamationen etc.

Highlights

Die Geschäftslogik zwischen Services bezeichnet man gewöhnlich als Workflows.

Die Flexibilität zur Änderung von Workflows ist eine der wichtigsten Stärken einer SOA.

Die Geschäftslogik zwischen Services bezeichnet man gewöhnlich als Workflows. Diese sind flexibler als die Geschäftslogik innerhalb von Services. Das müssen sie auch sein, da hier ein Großteil der heute erforderlichen Innovation und Produktivitätssteigerung stattfindet. Die Flexibilität zur Änderung von Workflows ist eine der wichtigsten Stärken einer SOA. In der Call-Center-Anwendung in unserem Beispiel sind die Workflows relativ einfach und standardisiert. Der wichtigste Workflow ist der Prozess, mit dem Anfragen an Second-Level-Support weitergeleitet, die Ergebnisse zurück an die Call-Center-Mitarbeiter gesendet und von diesen an den Kunden weitergegeben werden. Um die Kundenzufriedenheit zu verbessern, könnte unser fiktives Unternehmen GReP zum Beispiel entscheiden, dass bei Problemen mit hoher Priorität das Second-Level-Support-Team direkt mit dem Kunden zusammenarbeiten sollte. Eine solche Änderung im Workflow können auch Benutzer ohne IT-Kenntnisse mithilfe moderner portalbasierter Workflow-Funktionen selbst vornehmen.

Wenn GReP die neue Cross-Selling-Rolle für Call-Center-Mitarbeiter einführt, wird es sich dabei wahrscheinlich um eine separate modulare Anwendung handeln und nicht um eine Erweiterung des Workflows in der bestehenden Anwendung. Der Grund: Die neue Funktion ist enger mit anderen Verkaufsfunktionen im Unternehmen verknüpft und kann auf einfache Weise auf den zuvor entwickelten Funktionen für den Vertrieb aufbauen. Das SOA-Konzept, das eine solche Wiederverwendung von modularen Anwendungen möglich macht, nennt man „Vorlagen“ (Templates). Eine Vorlage beschreibt eine modulare Anwendung in abstrakter Form, einschließlich der verwendeten Services und der Art und Weise, wie diese über die Geschäftslogik miteinander verbunden sind. Bei einer Vorlage handelt es sich um eine XML-Datei, die Meta-informationen über eine Klasse ähnlicher modularer Anwendungen enthält. Sobald eine Vorlage definiert und gespeichert wurde, ist es – selbst für einen Benutzer ohne IT-Kenntnisse – relativ einfach, eine neue, zugehörige modulare Anwendung zu erstellen. Hierzu müssen lediglich die Parameter der Services und der Workflows zwischen ihnen geändert werden. In vielen Fällen können so die IT-Mitarbeiter entlastet werden, sodass sie neue, innovative Geschäftsfunktionen erstellen können. Natürlich gibt es auch andere Fälle, in denen neue Services oder eine komplexere Änderung der Geschäftslogik erforderlich sind. In diesen Fällen muss die IT-Abteilung mit einbezogen werden.

Highlights

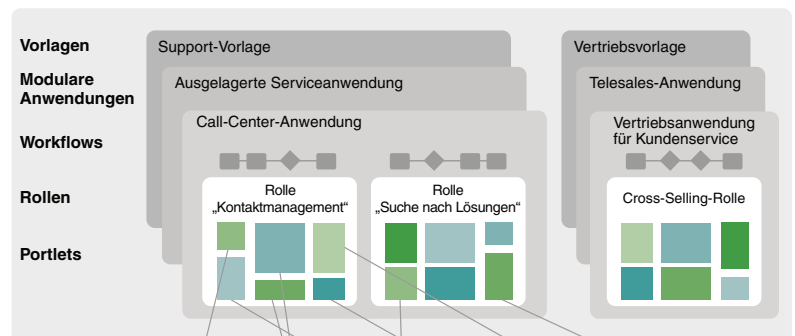
Die manuelle Übertragung von Informationen zwischen Anwendungen ist in die modularen Anwendungen und deren Workflows integriert. In früheren IT-Architekturkonzepten wurden die Verbindungen zwischen den Anwendungen und Komponenten auf IT-Ebene fest „verdrahtet“.

Die neue SOA ist gut strukturiert und abgestuft. Sie trennt die Geschäfts- und Benutzerperspektive von den unterstützenden IT-Komponenten.

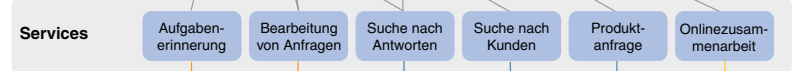
SOA-Systemarchitektur

Abbildung 4 zeigt die neue SOA-basierte Sicht der Call-Center-Umgebung und veranschaulicht die oben angesprochenen Aspekte bezüglich der Architektur. Von besonderem Interesse ist hierbei, dass diese neue Architektur gut strukturiert und abgestuft ist. Sie trennt die Geschäfts- und Benutzerperspektive von den unterstützenden IT-Komponenten. Die Geschäftsprozesse und die Flexibilität, sie zu ändern, werden so Teil der Benutzerperspektive – und die Endbenutzer werden produktiver und beweglicher, da sie das Anwendungsverhalten und den Anwendungsprozess an ihr Arbeitsmuster anpassen können. Die manuelle Übertragung von Informationen zwischen Anwendungen (wie in der Beschreibung der traditionellen Architektur in Abbildung 3 dargestellt) wurde in die modularen Anwendungen und deren Workflows integriert.

Benutzerperspektive



Geschäftsperspektive



IT-Perspektive



Abbildung 4. SOA-basierte Call-Center-Architektur (nicht alle Serviceverbindungen werden angezeigt)

Highlights

Dort kann die Informationsübertragung gesteuert werden, bleibt aber dennoch flexibel. In früheren IT-Architekturkonzepten – und auch beim Design der ursprünglichen Call-Center-Anwendung von GReP – wurden diese Verbindungen gewöhnlich zwischen den Anwendungen und Komponenten auf IT-Ebene fest „verdrahtet“.

Implementierung einer SOA mit IBM Produkten für Portale und Onlinezusammenarbeit

Nachdem die allgemeine Entscheidung für die Architektur gefallen war, folgten die Auswahl geeigneter Produkte und die Implementierung. Wie bereits erwähnt, gingen wir die Aufgabe aus der Sicht der Endbenutzer an. Die erste Frage lautete demnach: Wie können die oben beschriebenen Rollen und Services den Benutzern präsentiert werden? IBM WebSphere Portal Version 6.0, eine Software, die als Front-End einer SOA-Architektur konzipiert wurde, war die logische Wahl für die oben beschriebene Architektur.

Eine Rolle entspricht gewöhnlich einer Registerkarte oder einer Gruppe mehrerer Registerkarten in einem Portal. Grob gesagt kann jeder identifizierte Service einem Portlet zugeordnet werden.

Verwendung der IBM WebSphere Portal-Software

Abbildung 5 (siehe Seite 18) zeigt, wie die Services im Zusammenhang mit der Rolle „Kontaktmanagement“ in der Portalanzeige abgebildet werden. Eine Rolle entspricht gewöhnlich einer Registerkarte oder einer Gruppe mehrerer Registerkarten in dem Portal, und grob gesagt kann jeder identifizierte Service einem Portlet zugeordnet werden. In der in Abbildung 5 dargestellten Anzeige sehen wir jeweils ein Portlet, das den Services „Aufgabenerinnerung“ (oben links), „Suche nach Antworten“ (unten links), „Suche nach Kunden“ (oben rechts) und „Produktanfrage“ (unten rechts) zugeordnet ist.

Highlights

Was hier zuerst auffällt, ist das einheitliche Erscheinungsbild aller Portlets, obwohl deren Inhalte und Interaktionen zu drei sehr unterschiedlichen Anwendungen gehören. Die speziellen, ungleichen Client-Server-Schnittstellen wurden durch eine einheitliche Darstellung und Funktionsweise aller Elemente des Portals ersetzt. Die Benutzer verwenden künftig eine gemeinsame Schnittstelle zu allen Anwendungen, und die IT-Abteilung konnte die Kosten für Benutzerschulungen und -support deutlich senken.

Die ganze Ansicht ist deutlich übersichtlicher und einfacher als in dem traditionellen System, das in Abbildung 2 (siehe Seite 7) dargestellt wurde. Jeder Service hat seinen Platz, und Services, die in der Rolle „Kontaktmanagement“ nicht verwendet werden, sind ausgeblendet. Dennoch bietet das Portal den Benutzern die nötige Flexibilität, um die Anordnung der Elemente in der Anzeige zu ändern, sofern sie dies möchten. Der Umfang der erlaubten Anpassung kann vom Administrator des Systems auf der Basis der Anforderungen und Qualifikationen der Endbenutzer festgelegt werden. So können erfahrene Mitarbeiter bestimmte Portlets immer an Stellen anzeigen, die am besten zu ihrer Arbeitsweise passen, während neuen Mitarbeitern weniger Flexibilität eingeräumt wird, damit sie sich auf die Einarbeitung in die Anwendungen konzentrieren können.

Jeder Service hat seinen Platz in der Anzeige, und nicht verwendete Services sind ausgeblendet.



Abbildung 5. SOA-Kundenbetreuung, Rolle „Kontaktmanagement“

Highlights

Die beiden Portlets in der Mitte gehören beide zum Service „Bearbeitung von Anfragen“. Dieser Service verdeutlicht, dass eine exakte Übereinstimmung zwischen Service und Portlet nicht immer notwendig ist. Im Fall dieses Services ist die Menge an Informationen, die der Service bereitstellt, zu groß, um problemlos in ein einzelnes Portlet eingepasst zu werden. Also zeigt das Portlet oben in der Mitte (Current Work Item) die wichtigsten Informationen über den aktuellen Arbeitsvorgang an – Informationen, die immer benötigt werden. Über die drei Links am unteren Rand des Portlets kann der Benutzer auf Wunsch zusätzliche Informationen in einem zweiten Portlet anzeigen. Das unten in der Mitte dargestellte Portlet zeigt eine Liste der bisherigen Beschwerden des Kunden an, die vom Benutzer durch Anklicken des Links View Customer History aufgerufen wurde. Zu den weiteren anzeigbaren Optionen gehören die Schlüsselwörter, die das System für die Beschwerde ermittelt hat und die vom Benutzer editiert werden können (aufrufbar über den Link Keywords), und eine Liste ähnlicher Beschwerden (basierend auf den Schlüsselwörtern), die bereits abgeschlossen wurden (aufrufbar über den Link Show Similar Queries).

Diese drei alternativen Portlets verhalten sich wie Funktionen, die aus dem Portlet Current Work Item aufgerufen werden können. Sie wurden eigentlich als untergeordnete Services des Services „Bearbeitung von Anfragen“ implementiert. Sollte es jedoch notwendig werden, dass sie in größerem Umfang verwendet werden oder ihr Verhalten geändert wird, können sie zu vollwertigen Services in dieser Anwendung ausgebaut werden.

Dank der Integration zwischen Portlets können die Inhalte von Feldern in unterschiedlichen Portlets miteinander verknüpft werden, sodass die Informationen in einem Portlet automatisch auch in einem anderen zur Verfügung stehen. Damit entfällt das lästige Kopieren und Einfügen von Informationen durch die Endbenutzer.

Einer der offenkundigsten und wichtigsten Vorteile, den die WebSphere Portal-Software den Benutzern bietet, ist die Integration zwischen den Portlets. Dank dieser Integration können die Inhalte von Feldern in unterschiedlichen Portlets miteinander verknüpft werden, sodass die Informationen in einem Portlet automatisch auch in einem anderen zur Verfügung stehen. Damit entfällt das lästige Kopieren und Einfügen von Informationen durch die Endbenutzer. Wie Sie in Abbildung 5 sehen, sind der Name des Kunden und die Kundennummer zwischen dem aktuellen Arbeitsvorgang im CMS, der Liste bisheriger Beschwerden und den auf dem SAP-System basierenden Portlets für die Suche nach Kunden und die Abfrage des Gewährleistungsstatus synchronisiert. In gleicher Weise werden die von der CMS-Anwendung identifizierten Schlüsselwörter automatisch dem Portlet „Suche nach Antworten“ zur Verfügung gestellt.

Highlights

All diese Funktionen werden über die kooperativen Portlets und die Portlet-Verknüpfung implementiert, die mit WebSphere Portal Version 5 eingeführt wurden. Durch die Verknüpfung von Portlets, entweder programmgestützt oder per Definition in Web Services Description Language (WSDL), werden Änderungen an Eigenschaften in einem Portlet automatisch auf andere zugehörige Portlets übertragen. Je nach Rolle und Berechtigung des Benutzers kann dieser festlegen, in welchem Umfang Informationen zwischen Portlets übertragen werden. In unserem Beispiel war diese Funktion für die geschäftlichen Anforderungen der Endbenutzer jedoch nicht erforderlich und wurde daher auf die Anwendungsentwickler beschränkt.

Durch die Verknüpfung von Portlets, entweder programmgestützt oder per Definition in Web Services Description Language (WSDL), werden Änderungen an Eigenschaften in einem Portlet automatisch auf andere zugehörige Portlets übertragen.



Abbildung 6. SOA-Kundenbetreuung, Rolle „Suche nach Lösungen“

Highlights

Ein wichtiger Aspekt einer SOA ist die ausdrückliche Einbeziehung von Personen in den Geschäftsprozessen, wie in den IT-Systemen implementiert. Diese Integration bietet neue Möglichkeiten der Interaktion und beseitigt Engpässe in den Geschäftsprozessen, da sie die direkte Kommunikation zwischen zwei Personen erleichtert.

Für die in Abbildung 6 (siehe Seite 20) dargestellte Rolle „Suche nach Lösungen“ gelten dieselben Grundsätze. Diese Anzeige enthält drei derselben Portlets, die auch in der Rolle „Kontaktmanagement“ verfügbar sind. Darüber hinaus bietet der Service „Onlinezusammenarbeit“ drei zusätzliche Portlets: die beiden Portlets auf der rechten Seite – „People Finder“ für die Suche nach Personen und eine Kontaktliste, einschließlich mehrerer Möglichkeiten der Kommunikation mit einer Person (über ein mit der rechten Maustaste aufrufbares Kontextmenü) – und das Portlet für die E-Mail-Anzeige/Bearbeitung in der Mitte, das während der Arbeit in dieser Rolle in großem Umfang verwendet wird.

Ein wichtiger Aspekt einer SOA ist die ausdrückliche Einbeziehung von Personen in den Geschäftsprozessen, wie in den IT-Systemen implementiert. Diese Integration bietet neue Möglichkeiten der Interaktion und beseitigt Engpässe in den Geschäftsprozessen, da sie die direkte Kommunikation zwischen zwei Personen erleichtert (z. B. durch Instant Messaging). In unserem Szenario sorgt die Einführung der IBM Lotus Sametime-Funktionalität und deren enge Integration in die Portalumgebung für Vorteile in Form von besserer Zusammenarbeit und schnelleren Antworten. Im Mittelpunkt dieser Verbesserungen steht das Portlet „Kontaktliste“, das in Abbildung 6 rechts unten dargestellt ist. Die Liste zeigt auf visuelle Weise an, ob die Ansprechpartner des Benutzers online erreichbar sind. Diese Anwesenheitsanzeige wird auch in andere Portlets übertragen, wie Sie im Portlet „Aktueller Arbeitsvorgang“ (Current Work Item) in Abbildung 5 sehen. Zudem bietet sie den Call-Center-Mitarbeitern die Möglichkeit, die erreichbaren Personen im Fall eines dringenden Problems per Instant Messaging zu kontaktieren.

An dieser Stelle sollten wir eine neue Funktion in WebSphere Portal Version 6 erwähnen, auch wenn sie in unserem Szenario nicht vorkommt. Mit Version 6 wurde eine echte Workflow-Funktion innerhalb der Benutzerschnittstelle eingeführt. Beachten Sie, dass in der Rolle „Suche nach Lösungen“ ein impliziter Workflow enthalten ist, in dem der Call-Center-Mitarbeiter das Problem eines Kunden an Second-Level-Support weiterleitet und später eine Antwort erhält, die er dem Kunden zur Verfügung stellen muss. Mit der Workflow-Funktion in WebSphere Portal Version 6 können die Benutzer diesen Prozess und seine Entscheidungspunkte formell definieren, Aufgaben wie das Senden von E-Mails automatisieren und Arbeitsschritte im gesamten definierten Workflow verfolgen. Dank dieser neuen Funktion kann GReP sicherstellen, dass Probleme von Kunden nicht zwischen verschiedenen Abteilungen verloren gehen und die Anfragen von Kunden schneller bearbeitet werden, wenn der Call-Center-Mitarbeiter beschäftigt ist.

Highlights

Die erste Möglichkeit zur Beschaffung eines Portlets besteht darin, es zu leihen.

Portlets – leihen, kaufen oder selbst erstellen?

Nachdem wir uns für die WebSphere Portal-Plattform entschieden haben, untersuchen wir jetzt die verschiedenen Möglichkeiten der Beschaffung von Portlets. Es gibt im Prinzip drei Möglichkeiten, die alle in diesem Beispiel vertreten sind.

Die erste Möglichkeit besteht darin, ein Portlet zu „leihen“ – d. h. ein Portlet oder einen Service wieder zu verwenden, das/der bereits intern entwickelt wurde. In unserem Szenario trifft dies auf das Portlet „Suche nach Antworten“ zu. Diese Anwendung war zuvor als geringfügige Anpassung der WebSphere Content Discovery Server-Software entwickelt worden, wobei eine browserbasierte, portalfähige Schnittstelle verwendet wurde. Mit nur geringfügigen Änderungen, die eine Verknüpfung von Feldern ermöglichen sollten, wurde diese Anwendung als Portlet in der neuen Umgebung wieder verwendet.

Die zweite Wahl ist in der Regel der Kauf eines Portlets.

Die zweite Wahl ist in der Regel der Kauf eines Portlets. IBM vertreibt wie viele andere Anbieter portletfähige Schnittstellen zu einer Vielzahl verschiedener gängiger Anwendungen. In unserem Szenario gehören die Portlets für die Onlinezusammenarbeit zu dieser Kategorie. IBM Lotus Domino und erweiterte Produktportlets stellen eine Reihe sofort einsatzbereiter Portlets für die Onlinezusammenarbeit bereit. Dazu gehören Standardfunktionen von Lotus Notes und Domino wie E-Mail, Kalender und Terminplanung, Diskussion, Teamrooms und Aufgabenlisten sowie das Portlet Lotus Notes View, das den Zugriff auf Dokumente von jeder beliebigen Ansicht der Lotus Notes-Datenbank erlaubt. In unserem Szenario werden drei Portlets verwendet, die zu dieser Kategorie gehören.

Highlights

Das Portlet Lotus Notes View bietet den Benutzern Zugriff auf Lotus Notes-Datenbanken über einen Web-Browser, und das Portlet Lotus Sametime Contact List verschafft den Benutzern Zugang zur Instant-Messaging-Funktionalität von Lotus Sametime.

Die Integration von Anwesenheitsanzeige und Instant Messaging ist für unser fiktives Unternehmen von entscheidender Bedeutung, da diese neuen Tools die direkte Interaktion in Echtzeit zwischen den First- und Second-Level-Support-Teams ermöglichen und fördern.

Einbindung der IBM Lotus Produkte für die Onlinezusammenarbeit

Das Portlet Lotus Notes View (Abbildung 6, Mitte unten) verschafft den Benutzern Zugriff auf Lotus Notes-Datenbanken über einen Web-Browser. Dieses Portlet kann für die Bereitstellung des Zugangs zu bestimmten Arten von Lotus Notes-Datenbanken konfiguriert werden. In diesem Szenario wird das Portlet Lotus Notes View verwendet, um berechtigten Benutzern im Portal die Lotus Notes-Maildatenbank anzuzeigen. Das Portlet People Finder (Abbildung 6, oben rechts) kann verwendet werden, um nach Personen im Unternehmen zu suchen und detaillierte Informationen über sie anzuzeigen, z. B. Name, Telefonnummer und Funktion. Sobald die Person lokalisiert wurde, ist sie als Personenlink sichtbar, der die Onlineanwesenheit anzeigt. Außerdem wird ein Menü mit Instant-Messaging- und weiteren Optionen angezeigt. Das Portlet Lotus Sametime Contact List² (Abbildung 6, rechts unten) bietet den Benutzern die Möglichkeit, Instant Messages auf einem Lotus Sametime-Server mit allen Lotus Sametime-Benutzern im Unternehmen auszutauschen. Die Benutzer können an informellen Onlinegesprächen teilnehmen, nur Onlinenamen in der Liste anzeigen, die Listeneinträge sortieren, die Statusnachricht ändern oder Gruppen ändern.

Die Integration von Anwesenheitsanzeige und Instant-Messaging ist für unser fiktives Unternehmen von entscheidender Bedeutung, da diese neuen Tools die direkte Interaktion in Echtzeit zwischen den First- und Second-Level-Support-Teams ermöglichen und fördern. Dies trägt zu kürzeren Reaktionszeiten und höherer Kundenzufriedenheit bei. Dank der engen Verknüpfung der Lotus Sametime-Funktionalität und der auf WebSphere Portal basierenden Umgebung können die Benutzer auf einfache Weise zwischen E-Mail und Messaging wechseln, sofern erforderlich.

Ein weiterer wichtiger Trend in der Bereitstellung von Portlets ist die Erstellung anpassbarer Lösungen für bestimmte Problembereiche durch die Softwareanbieter. Indem die Anbieter mehrere Portlets in einer Lösung kombinieren, verringern sie den für die Implementierung nötigen Zeit- und Kostenaufwand. Ein gutes Beispiel hierfür ist IBM Workplace for SAP Software. Dabei handelt es sich um ein komplettes Angebot an rollenbasierten, anpassbaren Portlets für den Zugriff auf SAP-Funktionen, die Messung der Leistung im Vergleich zu definierten Zielen und die Verbindung mit einer Vielzahl von Funktionen für die Onlinezusammenarbeit, die von den Vertriebsmitarbeitern genutzt werden können.

Highlights

An letzter Stelle steht die Entwicklung eines Portlets, da es sich dabei um die teuerste Variante handelt. Allerdings können neue Portlets später von künftigen Systemen wieder verwendet werden, sodass in Zukunft Einsparungen bei den Kosten für die Anwendungsentwicklung möglich sind.

Am einfachsten ist die Erstellung eines Portlets, wenn die benötigten Services bereits über eine API zu einer traditionellen Anwendung verfügbar sind.

Schwieriger wird es bei einem System, das ursprünglich als voll integrierte Anwendung entwickelt wurde und in eigenständige, statusunabhängige und aufrufbare Komponenten zerlegt werden muss. Auf diese Komponenten können die Benutzer auf dieselbe Weise zugreifen wie auf Services, nämlich über portletbasierte Front-Ends.

Eine solche Lösung könnte die Basis für die Cross-Selling-Funktion bilden, die die Call-Center-Mitarbeiter nach dem Willen von GReP künftig ausüben sollen.

Die dritte Möglichkeit – die Entwicklung eines Portlets – steht an letzter Stelle, da es sich hierbei im Allgemeinen um die teuerste Variante handelt. Man muss jedoch bedenken, dass die Kosten nicht nur vom ersten Projekt getragen werden müssen. Neue Portlets können später von künftigen Systemen wieder verwendet werden, sodass in Zukunft Einsparungen bei den Kosten für die Anwendungsentwicklung möglich sind.

IBM WebSphere Portlet Factory

In diesem Prozess gibt es verschiedene Stufen der Komplexität, abhängig von Art und Ursprung der Services, auf die der Zugriff möglich sein soll. Am einfachsten ist es, wenn die benötigten Services bereits über eine API zu einer traditionellen Anwendung verfügbar sind. In unserem Beispiel trifft dies auf die SAP-Funktion zu. Das SAP-System bietet Business Application Programming Interfaces (BAPIs) für die Abfrage von Kunden- und Produktinformationen. IBM WebSphere Portlet Factory ist eine Software, die die nötige Funktionalität für die schnelle und einfache Erstellung von Portlets und die Interaktion mit den SAP-BAPIs bereitstellt. Wie in Abbildung 5 dargestellt, sind diese Portlets speziell auf die Anforderungen eines Call-Center-Mitarbeiters beim Zugriff auf das SAP-System abgestimmt. Die Benutzer müssen nicht die gesamte SAP-Schnittstelle erlernen und bei jeder Anfrage durch diese Schnittstelle navigieren. Mit diesen gezielt an ihre Erfordernisse angepassten Portlets können sie ihre Produktivität steigern und den Schulungsaufwand verringern.

Schwieriger wird es in unserem Beispiel beim Reklamationsbearbeitungssystem. Das CMS war ursprünglich als voll integrierte Anwendung entwickelt worden und musste daher in eigenständige, statusunabhängige und aufrufbare Komponenten zerlegt werden. Auf diese Komponenten können die Benutzer auf dieselbe Weise zugreifen wie auf Services, nämlich über portletbasierte Front-Ends. Auch in diesem Fall ist WebSphere Portlet Factory das Tool der Wahl für die schnelle und einfache Erstellung der Portlets. Doch der eigentliche Aufwand bestand darin, die alte CMS-Anwendung in ihre einzelnen Komponenten zu zerlegen und diese den zuvor identifizierten Services zuzuordnen.

Highlights

Beim Einsatz traditioneller Komponenten empfiehlt es sich, deren Umfang einzuschränken, sodass sie den Merkmalen des neuen, gewünschten Services so nahe wie möglich kommen.

Der Aufwand für die Zerlegung einer vorhandenen Anwendung in ihre einzelnen Bestandteile hängt von verschiedenen Faktoren ab, z. B. davon, in welcher Sprache die Anwendung codiert wurde, wie gut strukturiert sie war und inwieweit unterschiedliche Funktionen unabhängig von der Reihenfolge sind, in der sie aufgerufen werden. Im Allgemeinen lassen sich neuere Anwendungen, die in moderneren Sprachen geschrieben wurden, leichter in ihre einzelnen Komponenten aufteilen. Bei Verwendung der Funktionalität vorhandener Anwendungen als Basis für einen SOA-Service müssen eventuell Kompromisse bei der Definition des Services gemacht werden. Dies ist auf Einschränkungen bei Eingabe, Funktion oder Ausgabe der traditionellen Komponente zurückzuführen. Beim Einsatz solcher traditionellen Komponenten empfiehlt es sich, deren Umfang einzuschränken, sodass sie den Merkmalen des neuen gewünschten Services so nahe wie möglich kommen. Dies ist notwendig, damit die zugrunde liegende Funktion zu einem späteren Zeitpunkt so nahtlos wie möglich geändert oder ersetzt werden kann. Beachten Sie jedoch, dass eine solche spätere Migration nur durchgeführt werden sollte, wenn geschäftliche Anforderungen und der geschäftliche Nutzen es verlangen.

Zusammenfassung

Ein Unternehmen, das auf die in diesem White Paper beschriebene Art und Weise an die Implementierung einer SOA herangeht, verschafft sowohl den Benutzern als auch der IT-Abteilung eindeutige und klar definierte Vorteile. Wir wollen uns zunächst die neue Call-Center-Umgebung unseres fiktiven Unternehmens GReP aus der Sicht der Call-Center-Mitarbeiterin Maria ansehen.

Die neue Arbeitsoberfläche der Call-Center-Mitarbeiter ist deutlich besser strukturiert und organisiert. Dies trägt zur Steigerung der Effizienz und Verringerung des Einarbeitungsaufwands bei der Einführung neuer Funktionen bei.

Marias neue Arbeitsoberfläche, die wir in Abbildung 5 und 6 sehen, ist deutlich besser strukturiert und organisiert als zuvor. Funktionen, die sich auf eine bestimmte Gruppe von Aufgaben beziehen, sind zusammengefasst, für den Austausch von Statusinformationen integriert und logisch angeordnet. Nur die zu einem bestimmten Zeitpunkt tatsächlich benötigte Funktion wird angezeigt. Dagegen wurden Funktionen, die Maria nie braucht – z. B. die gesamte SAP-Schnittstelle, auf die sie früher Zugriff hatte – entfernt. Dadurch konnte Maria ihre tägliche Effizienz steigern und ihr Einarbeitungsaufwand bei der Einführung neuer Funktionen verringert werden.

Highlights

Die Effizienz dieser kontextbezogenen und gezielten Interaktion mit dem System verschafft dem Unternehmen einen weniger deutlich messbaren, aber dennoch vorhandenen Vorteil.

Die IT-Vorteile sind noch eindeutiger. Die Aufteilung der Funktion in ihre Komponenten erlaubt die Wiederverwendung von vorhandenem Code und die Senkung der Entwicklungskosten. Die lose Verbindung zwischen Komponenten vereinfacht die Implementierung und stellt sicher, dass die IT-Abteilung besser auf die Veränderungen reagieren kann, die das Unternehmen verlangt.

Die Rollen der Call-Center-Mitarbeiter sind jetzt in Form von Geschäfts-services definiert, die über kooperative Portlets miteinander verbunden und in einer einfachen Gruppe von Workflows zusammengefasst sind. Dadurch konnte die Produktivität von Maria und ihren Kollegen deutlich gesteigert werden. Außerdem ließ sich die Fehlerhäufigkeit im Zusammenhang mit der manuellen Eingabe oder dem Kopieren und Einfügen vorhandener Informationen zwischen Anwendungen reduzieren.

Die Effizienz dieser kontextbezogenen und gezielten Interaktion mit dem System verschafft dem Unternehmen einen weniger deutlich messbaren, aber dennoch vorhandenen Vorteil. Höher qualifizierte und erfahrene Benutzer wie Maria haben jetzt mehr Zeit, sich auf den Kunden – anstatt auf das IT-System – zu konzentrieren. Und diese verbesserte und erweiterte Interaktion schafft die Grundlage für Innovation.

Die IT-Vorteile sind noch eindeutiger. Die Aufteilung der Funktion in ihre Komponenten erlaubt die Wiederverwendung von bestehendem Code und die Senkung der Entwicklungskosten. Die lose Verbindung zwischen Komponenten vereinfacht die Implementierung und stellt sicher, dass die IT-Abteilung besser auf die Veränderungen reagieren kann, die das Unternehmen verlangt. Tatsächlich können entsprechend befugte Benutzer auch ohne IT-Kenntnisse Anwendungen mithilfe von Vorlagen selbst erstellen und anpassen, sofern vom Unternehmen benötigt. Dadurch werden Engpässe in der IT-Abteilung beseitigt, und die IT wird nicht mehr als Hindernis für Veränderungen wahrgenommen.

Da in diesem Beispiel die Integration eher an der Oberfläche abläuft und sich nicht auf tief greifende Prozess- und Datenintegration konzentriert, ging die Einführung des neuen Systems relativ schnell und kostengünstig vonstatten. Das bedeutet jedoch nicht, dass auf eine tiefer gehende Integration verzichtet werden kann oder sollte. Andere Prozesse erfordern andere Ansätze, abhängig davon, wie eng die gewünschte Integration ist. Dieses geringere Maß an Integration kann aber schon sehr erfolgreiche Ansätze liefern.

Highlights

Unser fiktives Unternehmen hat die ersten Schritte auf dem Weg zur Implementierung einer serviceorientierten Architektur getan. Es hat den Nutzen einer SOA unter Beweis gestellt und gezeigt, wie die Benutzer ihre täglichen Aufgaben produktiver und innovativer erledigen können.

Da die IT-Mitarbeiter weniger Zeit auf die Back-End-Integration verwendet haben, konnten sie sich vorrangig auf neue Funktionalität und Middleware konzentrieren, von der das Unternehmen direkt profitiert. Dank der Anwesenheitsanzeige und Interaktion in Echtzeit zwischen den First- und Second-Level-Support-Teams, ermöglicht durch Lotus Sametime und die zugrunde liegende Infrastruktur, können von Kunden gemeldete neue Probleme schneller und zur größeren Zufriedenheit der Kunden bearbeitet werden. Und auch wenn wir sie hier nicht ausführlich beschrieben haben, werden die integrierten Workflow-Funktionen in WebSphere in Zukunft zu weiteren Effizienzen beitragen.

GReP konzentrierte sich zunächst auf die Frage, welche Geschäftsservices das Call-Center benötigte und wie diese Services mit IBM Produkten für Portale und Onlinezusammenarbeit auf einfache Weise integriert und den Benutzern zur Verfügung gestellt werden konnten. Damit hat GReP bereits die ersten Schritte auf dem Weg zur Implementierung einer serviceorientierten Architektur getan. Diese ersten Schritte wirken sich innerhalb kürzester Zeit positiv auf das Unternehmen aus – sowohl auf die Geschäftsbereiche als auch die IT-Abteilung. Sie stellen den Nutzen einer SOA unter Beweis und zeigen, wie die Benutzer ihre täglichen Aufgaben produktiver und innovativer erledigen können. Von diesem Punkt aus kann GReP den nächsten, umfangreicheren Prozess in Angriff nehmen – die Arbeitsweise seiner Mitarbeiter neu zu definieren. Dabei kann das Unternehmen sicher sein, dass es den Prozess genau versteht und dass die ausgewählte Middleware die geschäftlichen Veränderungen ermöglicht, die das Unternehmen braucht. Ein altes irisches Sprichwort sagt: Tús maith, leath na hoibre – ein guter Anfang ist schon die halbe Arbeit.

Weitere Informationen

Wenn Sie mehr über die Vorteile einer SOA für Ihr Unternehmen erfahren möchten, besuchen Sie:

ibm.com/soa/de

Weitere Informationen über eine benutzerorientierte Herangehensweise an eine SOA finden Sie unter:

ibm.com/software/de/lotus



IBM Deutschland GmbH
70548 Stuttgart
ibm.com/de

IBM Österreich
Obere Donaustraße 95
1020 Wien
ibm.com/at

IBM Schweiz
Vulkanstrasse 106
8010 Zürich
ibm.com/ch

Die IBM Homepage finden Sie unter:

ibm.com

IBM, das IBM Logo und ibm.com sind eingetragene Marken der IBM Corporation.

Domino, Lotus, Lotus Notes, Notes, OmniFind, Sametime, WebSphere and Workplace sind Marken der IBM Corporation in den USA und/oder anderen Ländern.

Weitere Unternehmens-, Produkt- oder Servicennamen können Marken anderer Hersteller sein.

Vertragsbedingungen und Preise erhalten Sie bei den IBM Geschäftsstellen und/oder den IBM Business Partnern. Die Produktinformationen geben den derzeitigen Stand wieder. Gegenstand und Umfang der Leistungen bestimmen sich ausschließlich nach den jeweiligen Verträgen.

Der Inhalt dieser Dokumentation dient nur zu Informationszwecken. Obwohl die in dieser Dokumentation enthaltenen Informationen auf ihre Vollständigkeit und Genauigkeit hin überprüft wurden, wird sie auf der Grundlage des gegenwärtigen Zustands (auf „as-is“-Basis) ohne jegliche Gewährleistung zur Verfügung gestellt. Darüber hinaus basieren diese Informationen auf der aktuellen Produktplanung und -strategie von IBM, die sich jederzeit ohne Vorankündigung ändern kann. IBM übernimmt keine Haftung für irgendwelche Schäden, die aus der Nutzung dieser oder einer anderen Dokumentation entstehen oder damit in Zusammenhang stehen. Aus dem Inhalt dieser Dokumentation können kein Gewährleistungsanspruch oder andere Anforderungen an IBM (oder seine Lieferanten oder Lizenzgeber) abgeleitet werden, noch kann der Inhalt eine Änderung der Bedingungen der geltenden Lizenzvereinbarung, der die Nutzung der IBM Software unterliegt, bewirken.

Diese Veröffentlichung enthält Internetadressen von anderen Herstellern als IBM. IBM übernimmt keinerlei Verantwortung für die auf diesen Websites enthaltenen Informationen.

-
- 1 IBM WebSphere Content Discovery Server basierte auf der iPhrase OneStep-Plattform, die IBM 2005 aufkaufte, und wird jetzt als IBM WebSphere OmniFind Discovery Edition bezeichnet.
 - 2 Frühere Bezeichnung: Lotus Instant Messaging Contact List.

© Copyright IBM Corporation 2006
Alle Rechte vorbehalten.

LOW10770-DEDE-00
IBM Form GM12-6825-00 (02/2007)