

WCMS und Portal „richtig“ integrieren

In der Werbung klingt es so einfach: Ein Portal integriert alle in einer Organisation bestehenden Anwendungen und stellt diese den Nutzern rollenbasiert in einer Nutzer-Oberfläche zur Verfügung. In der Realität birgt die Integration jedoch Probleme, vor allem, wenn es um das in der Regel bereits bestehende Web-Content-Management-System geht. Dieses wartet genauso mit einer eigenen Navigationsstruktur auf wie das Portal. Soll der Web-Redakteur beide Navigationsstrukturen parallel pflegen? Dies ist nicht die einzige Frage bei der Integration von Portal und WCMS.

Thomas Radigewski

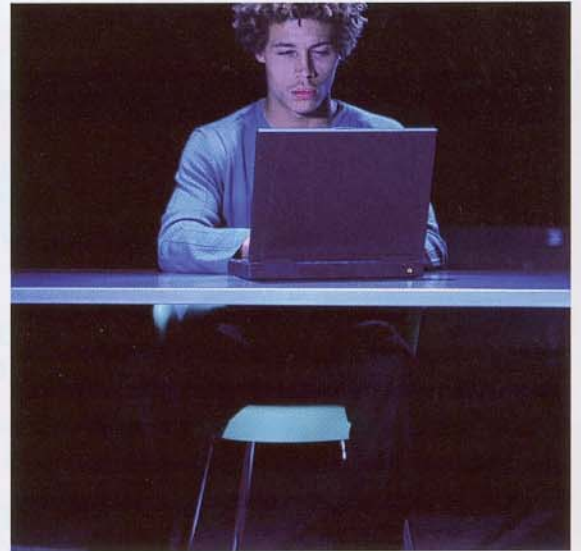
FAST ALLE Unternehmen oder Behörden, die heute ein Portal einführen wollen, setzen bereits ein Web-Content-Management-System ein, das eine meist dynamische Navigationsstruktur vorgibt. Das Portal selbst integriert alle vorhandenen Anwendungen wie etwa SAP-Applikationen, relationale Datenbanken, CRM-Systeme, Lotus-Notes-Datenbanken, selbst entwickelte Portlets sowie redaktionelle Inhalte. Diese Inhalte, die im Portal an den verschiedensten Stellen veröffentlicht werden, sollen aus Effizienzgründen auch nach der Integration weiterhin zentral im WCMS gepflegt werden.

Seitenaufbau im WebSphere-Portal bei der WCMS-Integration

Bei der Integration eines Web-Content-Management-Systems in das IBM-WebSphere-Portal „liefert“ das Portal die in der Regel ein- oder zweizeilige Head-Navigation, aus der jeder Hauptnavigationspunkt auf eine eigene Portalseite verweist. Der untere Bereich der Portalseite speist sich meist aus dem WCMS: Üblich ist eine Aufteilung in Navigation auf der linken Seite und Content auf der rechten Seite. Die Navigation wird dabei von einem Navigations-Portlet „erzeugt“, der Content von einem Standard-Content-Portlet.

Doppelte Pflegeaufwände

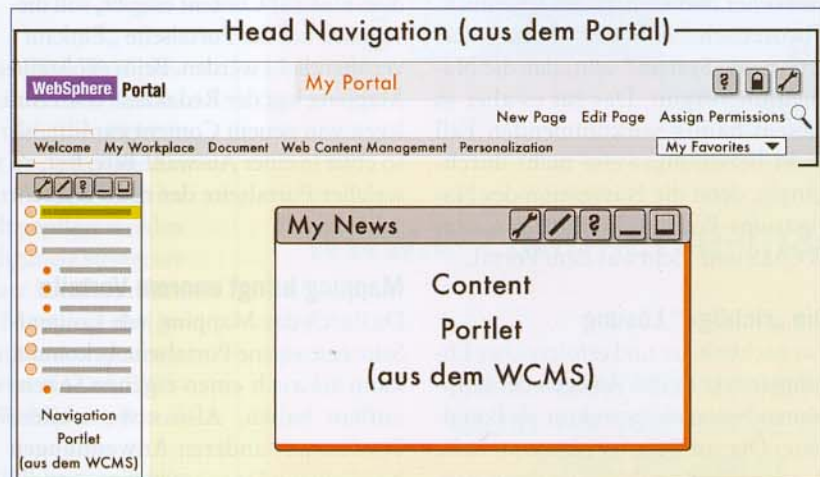
Will ein Redakteur bei diesem „gegebenen“ Seitenaufbau einen neuen Content-Bereich erstellen, so ist dieser sowohl im Portal als auch im WCMS anzulegen. Damit muss die gleiche Navigationsstruktur in zwei Systemen gepflegt werden. Dies gestaltet sich nicht nur unnötig zeitaufwändig, sondern ist auch fehlerträchtig. Ein weiteres Problem: Durch den Einsatz der von WCMS-Herstellern mitgelieferten Standard-Portlets wird aller Content in der gleichen Portalseite angezeigt. Durch diesen Umstand ist der Aufbau der Portalseite festgelegt. Die Konsequenz: Auf einer einzelnen Content-



Seite des Portals können keine zusätzlichen Portlets angezeigt werden, ohne dass diese auf allen aus dem WCMS befüllten Content-Seiten des Portals sichtbar sind.

WebSphere-Portal: nur vordergründige Lösungsmöglichkeit

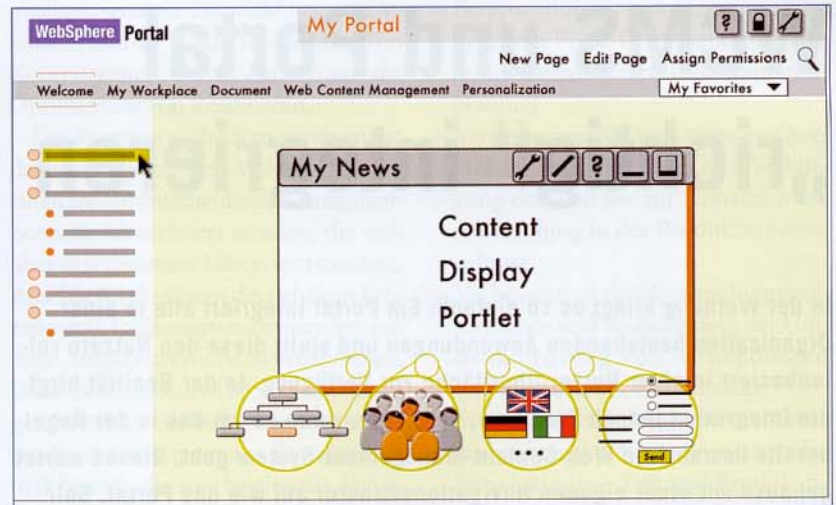
Das erste Problem lässt sich im WebSphere-Portal gar nicht, das zweite nur vordergründig „lösen“: Soll etwa aus dem Content auf ein elektronisches Feedback-Formular (oder jede andere als Portlet realisierte Anwendung) verlinkt werden, so müsste das Formular eigentlich in der linken Navigationsleiste erscheinen. Eigentlich,



Unflexibel: der „starre“ Aufbau einer Portalseite mit Standard-Portlets.

Content Management | PORTALE

Über intelligente Content Display Portlets werden redaktionelle Anforderungen wie Mehrsprachigkeit, Personalisierung, Einbindung von Formularen oder Link Management erfüllt. Wichtig: Die linke Navigation kommt bei dieser Lösung ebenfalls aus dem Portal.



denn dies ist aufgrund der beschriebenen Problemstellung nicht möglich. Also öffnet sich beim Klicken des Feedback-Formulars ein entsprechender Head-Navigationspunkt des Portals als neue Portalseite. Diese beinhaltet dann ein entsprechendes Navigations-Portlet, ein Standard-Content- und ein Feedback-Portlet. Tech-

Navigationen-Portlet mehr, sondern eine Navigation, die aus dem Portal gespeist wird. Jede Content-Seite erhält damit sozusagen eine eigene Portalseite. Dafür müssen die Content Portlets jedoch „wissen“, welchen Content sie konkret anzeigen sollen. Das ist dann möglich, wenn ein Mapping für die gesamte Content-Struktur des WCMS eingeführt wird. Mit diesem Mapping wird im WCMS jedem Content genau die Portalseite zugewiesen, auf der er erscheinen soll.

ren Layout erscheinen. Damit ergeben sich für den Redakteur neue Freiheitsgrade, weil er den Aufbau der Seite den Bedürfnissen des jeweiligen Contents anpassen kann. Aber auch der Portalnutzer profitiert: Ihm werden zusätzliche Anwendungen angeboten, ohne dass er beim Klicken in eine andere Navigationsstruktur „umgeleitet“ werden muss.

Wie Portale ihren Nutzen wirklich entfalten

nisch machbar, jedoch von der Navigationsstruktur her nicht stringent: Der Nutzer der ursprünglichen Portalseite ist nach dem Aufruf des Formulars auf einer anderen Portalseite als vorher und verliert den Überblick. Theoretisch soll das Portal das „führende System“ sein, das die Navigation vorgibt. Das tut es aber in diesem häufig vorkommenden Fall nicht beziehungsweise nicht durchgängig, denn die Navigation des Navigations-Portlets kommt aus dem WCMS und nicht aus dem Portal.

Die „richtige“ Lösung

Ein nachhaltiger und erfolgreicher Lösungsansatz ist das Anlegen der kompletten Navigationsstruktur als Portalseite. Der im Beispiel genannte linke Bereich der Portalseite ist dann kein

Implizites und explizites Content Mapping

Das Mapping der WCMS-Content-Struktur lässt sich implizit wie explizit organisieren. Beim impliziten Mapping erfolgt die Zuordnung von neuem Content auf der Basis vorgegebener Regeln. Wenn etwa ein Redakteur aus dem Einkauf Content eingibt, soll dieser auch auf der Portalseite „Einkauf“ veröffentlicht werden. Beim expliziten Mapping legt der Redakteur beim Anlegen von neuem Content explizit, also etwa in einer Auswahl-Box, fest, zu welcher Portalseite der neue Content gehören soll.

Mapping bringt enorme Vorteile

Da durch das Mapping jede Content-Seite eine eigene Portalseite bekommt, kann sie auch einen eigenen Seitenaufbau haben. Also etwa weitere Portlets mit anderen Anwendungen beinhalten oder sogar in einem ande-

Zusatznutzen von Content Display Portlets

Außer hoher Flexibilität bei der Gestaltung von Content-Seiten im Portal haben die Redakteure aber weitere Anforderungen für das Veröffentlichen von Content, beispielsweise Mehrsprachigkeit, Personalisierung oder Link-Management. Auch diese Anforderungen lassen sich mit den Standard-Content-Portlets der WCMS nicht lösen. An dieser Stelle

THOMAS RADIGEWSKI

ist Gründer und Geschäftsführer von RADIGEWSKI Informatik. Aufgrund seiner langjährigen Erfahrung in der Software-Entwicklung und Projektrealisierung von WCMS- und Portal-Projekten auf der Basis von IBM-Technologie wurde er von Big Blue im vergangenen Jahr ins Autoren-Team des RedBooks „IBM Workplace Web-Content-Management-System 5.1“ berufen. Er hält außerdem regelmäßig Fachvorträge auf IBM-Veranstaltungen.

hilft die Entwicklung intelligenter Content Display Portlets weiter, die über die J2EE-API des anzubindenden WCMS auf dessen Repository zuzugreifen. Mit solchen Content Display Portlets lassen sich zusätzliche Vorteile realisieren:

- > Da nun die Navigation des Portals vollständig aus diesem kommt, kann auch die Verwaltung der Leserechte komplett über das Portal abgewickelt werden. Damit lässt sich beispielsweise festlegen, dass bestimmte Inhalte nur von der Personalabteilung, vom Einkauf usw. gelesen werden können.
- > Da die oben erwähnten intelligenten Content Display Portlets auf das Repository des WCMS zugreifen und für jedes Standard-Content-Portlet den passenden Content aus dem WCMS finden, ist es auch möglich, Content problemlos mehrsprachig im Portal anzuzeigen, wenn die mehrsprachigen Inhalte im WCMS hinterlegt sind. Ohne den Einsatz von Content Display Portlets ist die Pflege mehrsprachiger Inhalte nur über jeweils separate Webseiten möglich.
- > Standard-Content-Portlets zeigen Inhalte von verlinkten Seiten auf der gleichen Portalseite mit der gleichen Navigation an, auch wenn deren Inhalte in der Struktur an anderer Stelle eingehängt sind. Beispiel: Eine Pressemitteilung, die in der Navigationsstruktur unter „Pressecenter“ eingehängt ist, verlinkt auch auf die Wegbeschreibung, die in der Navigationsstruktur unter „Unternehmen“ steht. Werden Content Display Portlets eingesetzt, „switcht“ die Navigation bei Klick auf den verlinkten Inhalt im Beispiel auch auf „Unternehmen“.
- > Links in Content-Seiten müssen beim Einsatz von Content Display Portlets nicht mehr zwingend ausschließlich auf andere Content-Seiten verlinken, sondern können dies auch auf Seiten ohne Content tun, so zum Beispiel auf Formulare oder andere Anwendungen.
- > Und endlich arbeiten die Redakteure ausschließlich im WCMS, wo sie über die Content Display Portlets alle für die Veröffentlichung des jeweiligen Inhalts im Portal relevanten Informationen eingeben. Eine doppelte Pflege von WCMS- und Portal-Navigationsstruktur entfällt.

Fazit

Die Standard-Content-Portlets der in das IBM-WebSphere-Portal zu integrierenden WCMS gewähren den Redakteuren nach der Integration des WCMS nicht die notwendige Flexibilität bei der Gestaltung der Portalseiten. Sie führen außerdem dazu, dass sowohl im WCMS wie im Portal die Navigationsstrukturen doppelt gepflegt werden müssen und tragen durch eine nicht durchgängig stringente Navigation zur Verunsicherung der Nutzer bei. Diese Probleme lassen sich durch intelligentes Mapping der im WCMS abgelegten Inhalte und den Einsatz von smarten Content Display Portlets, wie sie etwa von RADIGEWS-KI Informatik und WP-Experts für das WebSphere-Portal entwickelt wurden, dauerhaft lösen.

URL: www.radi.info