



smart.View 1.00

Das einfache Werkzeug für die Integration von Notes-Anwendungen in Ihr Intranet – Notes-Ansichten die nicht nach Lotus Notes aussehen



17.04.2008

Konfigurierbare Webanwendungen

Die Ausgangssituation

„Wir sind verkauft !! – Ab dem 31.12. gehören wir einem neuen Eigentümer“.

Dies ist nur eine von zahlreichen Neuigkeiten welche einem Fachbereich in jedem Unternehmen das Blut in den Adern gefrieren lässt.

- Wie verhält sich der neue Eigentümer?
- Was sind seine Erwartungen oder gar Zielvorgaben?
- Was hat das für Auswirkungen auf unsere Systeme und die IT-Infrastruktur?
- Und was passiert mit dem Frontend dass ich nun mit so viel Liebe zum Detail an unser CI angepasst habe?

Aber auch wesentlich weniger „dramatische“ Veränderungen erfordern einen erheblichen Eingriff in bestehende Systeme. Eine neue Ansicht, eine neue Schaltfläche oder die Änderung der Sortierreihenfolge sind zumeist mit Anpassungen an der Gestaltung meiner Datenbank verbunden. Unterliegt mein Unternehmen externen Dokumentationspflichten (wie z. B. Sox) ist jede noch so kleine Anpassung mit erheblichen Test- und Dokumentationsaufwänden verbunden.

Solchen Veränderungen vorbeugend zu begegnen ist die eigentliche Herausforderung für einen (internen) IT-Dienstleister heutzutage.

Mögliche Szenarien

Um auf neue Anforderungen der Endkunden schnell und kompetent reagieren zu können stehen einem Anbieter zwei grundverschiedene Szenarien zur Verfügung:

- 1) die Bereitstellung einer umfassenden Support-Verfügbarkeit (meist verbunden mit hohem Personalaufwand und den damit verbundenen Kosten)
- 2) Verwendung von (globalen) Standards, insbesondere Multi-Tier-Architekturen

Angesichts der vorherrschenden Budget-Restriktionen wollen wir für den weiteren Verlauf die Option (1) geflissentlich ignorieren.

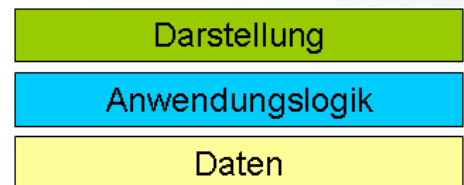
Betrachten wir das zweite Szenario unter Verwendung von Lotus Domino:
Klassischerweise wird unter dem Begriff „Multi-Tier-Architektur“ die Trennung von Datenhaltung, Anwendungslogik und Darstellung verstanden.

Durch den in Lotus Domino integrierten Web-Server gibt es für die Ansicht der Anwendung zwei unterschiedliche Fenster: Den Lotus Notes-Client (1) und einen Web-Browser (2):

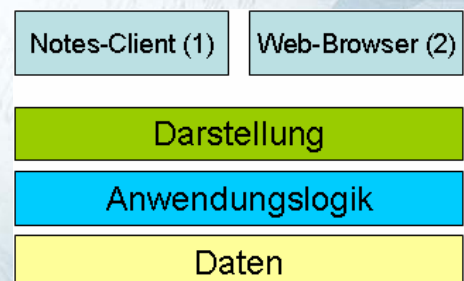
Während der Notes-Client ein wesentlich breiteres Portfolio an



Beispiel einer Standard Notes-Ansicht im Web



Multi-Tier-Architektur



Darstellung bei Web-Anwendungen unter Lotus Domino



smart.View 1.00

Das einfache Werkzeug für die Integration von Notes-Anwendungen in Ihr Intranet – Notes-Ansichten die nicht nach Lotus Notes aussehen



17.04.2008

Funktionalitäten (Verarbeitung von Datei-Anhängen, Integration von Office-Funktionalitäten, ...) anbietet ist er im Hinblick auf tiefgreifende Veränderungen in den bestehenden IT-Systemen eher ein Risiko.

Bei z. B. einem Wechsel der verwendeten Messaging-Infrastruktur führen Anwendungen welche ausschliesslich auf dem Notes-Client aufbauen zu erheblichen Mehrkosten in punkto Administration und Hardware-Anforderungen.

Verwendet man für die Bedienung von Anwendungen auf einem Domino-Server ein Browser-Frontend können die Vorteile von Lotus Domino (Zugriffs-Steuerung, Workflows, ...) mit den weitestgehend flexiblen Gestaltungsmöglichkeiten in einem Webfrontend kombiniert werden.

Hintergründe, Schrift-Farben und andere Gestaltungselemente lassen sich mit standardisierten Elementen wie CSS und HTML beliebig ausschmücken und darstellen.

Im folgenden wird anhand von Ansichten demonstriert wie man Domino-Anwendungen konzipieren kann welche nicht nach Domino aussehen:

Architektur für Web-Anwendungen

Bei der Darstellung von Formularen und Ansichten im Browser-Frontend übernimmt der Domino-Server die Überführung der Notes-Elemente und Inhalte in HTML.

Die hierbei generierten HTTP-Seiten haben etliche „Schönheitsfehler“ welche zu sehr ärgerlichen Effekten führen können:

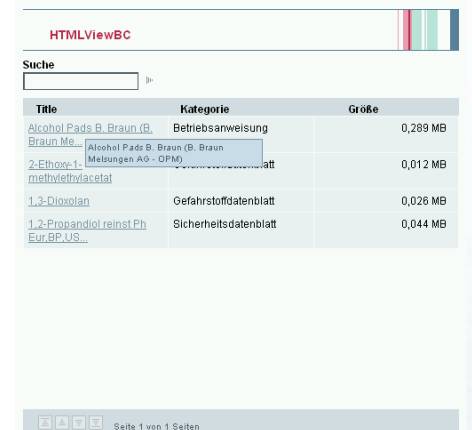
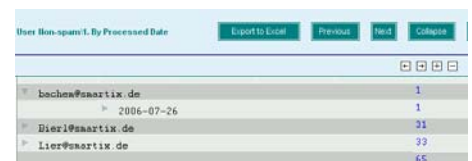
- Verwendung von Java-Applets: Aktionen in Ansichten, die Navigation und RichText-Felder werden in der Regel mit Java-Applets dargestellt welche nicht auf jedem Rechner fehlerfrei funktionieren (z. B. Zugriff auf das Mailfile aus dem Internetcafe)
- Tabellenformatierungen (insbesondere Rahmen) werden für jede Tabelle einheitlich gestaltet – unabhängig von den in Notes verwendeten Formatierungen
- Aktionen sehen in der Regel sehr unschön aus
- Und vieles mehr...

Werden in einer Web-Anwendung mehrere Ansichten verwendet werden in der Regel anstatt eines „globalen“ `$$ViewTemplate` für jede Ansicht eigene „`$$ViewTemplates`“ erstellt.

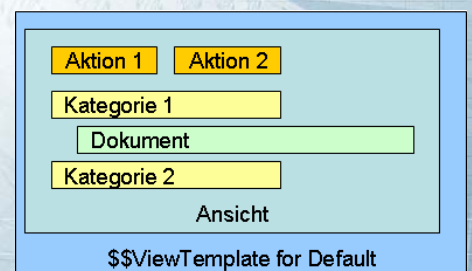
Dies bläht zum einen die verwendete Gestaltung auf und erschwert zum Anderen die Wartbarkeit bei Änderungen in der verwendeten Farbgebung.

Werden Anwendungen sowohl von Notes- wie auch Web-Clients verwendet führt dies auch häufig zu einer doppelten Anzahl von Ansichten (für den Web- und Notes-Zugriff).

Technisch ist es jedoch nur relevant dass die Dokumente geöffnet werden können, Kategorien sich erweitern lassen und die in der Aktionsleiste verwendeten Aktionen funktionieren.



Beispiele für Standard.Notes-Ansichten im Web



Typischer Aufbau einer Notes-Ansicht im Web-Browser



smart.View 1.00

Das einfache Werkzeug für die Integration von Notes-Anwendungen in Ihr Intranet – Notes-Ansichten die nicht nach Lotus Notes aussehen



17.04.2008

Ein ideales Architektur-Konzept für Ansichten ergibt sich daher wie folgt:

- a) Alle Ansichten sind für Notes- und WebClients „sichtbar“
- b) Beim Web-Zugriff werden diese in ein „globales \$\$ViewTemplate“ – und bei der Suche in ein „globales \$\$SearchTemplate“ eingebunden
- c) Die Gestaltung / Darstellung der Ansicht ergibt sich aus CSS-Informationen welche über ein mit dem Ansichten-Namen verbundenes Profil-Dokument angezogen werden.
- d) Diese globalen Vorlagen werden in Bereiche unterteilt die in Abhängigkeit der hinterlegten Konfiguration angezeigt und formatiert werden (z. B. Einblenden einer Suchleiste)
- e) Der verarbeitende Code für die Transformation des Layouts von „Standard-Domino“ zu einem „schönen Web-Frontend“ wird über eine JS-Steuerung beim Aufruf der Ansicht vorgenommen.

Ruft man das Formular „\$\$ViewTemplateDefault“ aus Notes heraus auf können in diesem CSS-Informationen eingegeben und diese Ansichten zugewiesen werden (z. B. der Ansicht: Calendar, Billing extern...)

Wird nun die entsprechende Ansicht im Web-Frontend aufgerufen wird über den Schlüssel „Ansichts-Name“ das entsprechende Layout (CSS)-Profil angezogen.

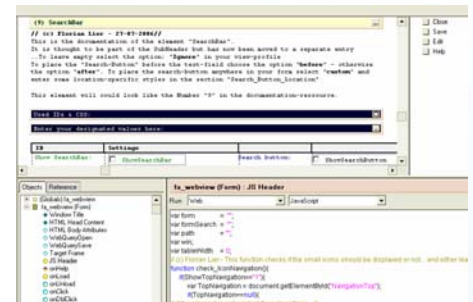
Ein im Benutzerfrontend ablaufender JS-Code (siehe Beispiel 1) im „onLoad“-Event der Maske weist den von Domino gerenderten HTML-Inhalten (Aktions-Leiste, Suchleiste, Kategorien) die hinterlegten Darstellungs-Informationen im CSS-Format zu.

Neben der Darstellung können über die in JS zur Verfügung stehenden Technologien (Tabellen erweitern und modifizieren) auch zusätzliche Aktionen zum editieren oder löschen von Dokumenten in die Ansichten mit aufgenommen werden.

Sind solche „Erweiterungen“ Bestandteil einer globalen Schablone, fallen Tests und Dokumentationen für die Erweiterung von bestehenden Anwendungen minimal aus, da bereits evaluierte Technologien und Funktionen nur „eingebledet“ werden.

Bezogen auf die von Lotus Domino automatisch generierten HTML-Elemente lässt sich hierbei ebenfalls eine „Multi-Tier-Architektur“ verwenden.

Lotus Domino erstellt für vorhandene Schaltflächen HTML-Schaltflächen. Verbirgt man diese Elemente für die Endanwender – kann man diesen stattdessen ein „schön formatiertes“ HTML-Frontend zur Verfügung stellen in welchen die Domino-Elemente beim Betätigen der Schaltfläche angesprochen und ausgeführt werden.



Optimierter Aufbau für ein Web-Rendering von Notes-Anwendungen



Zuweisung von CSS-Informationen an Ansichten



smart.View 1.00

Das einfache Werkzeug für die Integration von Notes-Anwendungen in Ihr Intranet – Notes-Ansichten die nicht nach Lotus Notes aussehen



17.04.2008

Auf diese Weise lassen sich Domino-Standards mit eigenen „Firmen-Standards“ für die Steuerung von Domino-Anwendungen kombinieren.

Fazit

Berücksichtigt man bei der Planung und Umsetzung von Anforderungen unter Lotus Domino das Webfrontend als möglichen Client und investiert man im Vorfeld in den Aufbau eines effektiven Web-Frameworks erzielt man bei Änderungsanforderungen erhebliche Einsparungspotentiale. Änderungen sowohl im Look und Feel als auch bei der Umsetzung eines zusätzlichen Webfrontends lassen sich so erheblich billiger gestalten und innerhalb kürzester Zeit umsetzen.

Die in diesem Artikel beschriebene Lösung kann auf der Webseite der GMI KG unter:

<http://www.gesmi.de>

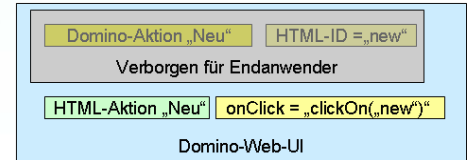
in dem Bereich „Tools – ViewTemplate“ heruntergeladen werden.

Veit Florian Lier

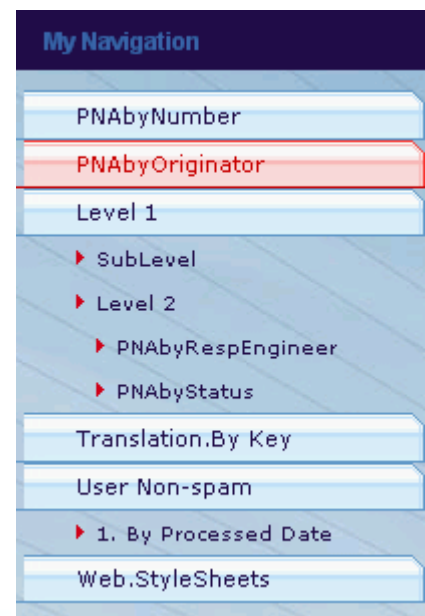
Veit Florian Lier ist seit 2 Monaten bei der GMI KG als „Chief Technical Architect“ tätig. In diesem Rahmen befasst er sich mit der Planung und Umsetzung von innovativen Produkten für die Prozess-Optimierung und Anwendungs-Migration im Umfeld von Lotus Domino.

System requirements :

Smart.View läuft mit Lotus Notes Release 5, 6, 7 und 8.



Integration von Firmen- mit Domino-Standards



Navigation created by smart_View