

Domino und XML

XML hat sich in Unternehmen einen Platz etablieren können, mit dem in vielen Bereichen XML „under the cover“ still seinen Dienst verrichtet, um beispielsweise Daten von einem System für ein anderes System bereit zu stellen. Es gibt heute eine Vielzahl von Anwendungen, welche auf der Technologie von XML basieren. Nehmen wir alleine einmal die WebSphere Integration auf den unterschiedlichen Plattformen – ein nicht unwesentlicher Faktor der Umsetzung dieser Funktionen basiert auf XML-Beschreibungen und -Dokumenten, mittels deren die Integration realisiert wird.

Domino und Notes bieten seit Jahren ebenfalls die Möglichkeit, XML einzubinden. Damit lassen sich die Einsatzbereiche von Domino elegant erweitern. Dabei steht es den Anwendungsentwicklern frei, XML in Dokumenten oder Ansichten einzubinden. Während die Integration von XML in Masken relativ einfach umgesetzt werden kann, sind bei der XML-Integration in Ansichten durchaus komplexe Schritte und Vorgehen anzuwenden, welche zum Ziel führen. Funktionen, die in der Basis des Domino Designers enthalten sind, unterstützen bei der XML Entwicklung genauso, wie LotusScript Funktionen und Befehle als auch die in Java zur Verfügung stehenden Bausteine für die XML Integration.

Es ist schon bezeichnend, wie stark die Struktur einer Domino Datenbank mit den XML-Grundsätzen vergleichbar ist. Die Trennung der Daten in Form von Dokumenten von der Gestaltung in Form von Masken bietet sich ideal dazu an, eine XML-Umsetzung dieser Komponenten durchzuführen, welche sich ideal in XML Fragmente übertragen lassen. XML arbeitet, ähnlich wie Domino auch, auf Basis einer Trennung der einzelnen Elemente bzw. der Ebenen.

Aufgrund der Besonderheiten von Domino hat man ein spezielles XML Derivat entwickelt – das so genannte DXL. Mit DXL lassen sich Domino Datenbanken in XML beschreiben. Diese „Sprache“ ist besonders für die Gestaltungsbereiche einer Datenbank ideal in der Anwendung. Damit lassen sich die Definitionen einer Domino Datenbank mittels XML exportieren und zum Beispiel mit einer speziellen Transformation in ein anderes Format umsetzen. Sollten Sie sich nun fragen, wozu man eine solche Umwandlung benötigt, dann denken Sie ganz einfach einmal daran, wie sich damit die Integration von HTML Anweisungen in Domino realisieren lassen könnte. Mit einem Klick eine HTML Maske in Domino umzusetzen, um dann damit die enthaltenen Informationen Domino-like zu verarbeiten, ist einer der Vorteile, die sich mit XML bieten. Mit den XML Komponenten stehen und also Import- und Exportfunktionen zum Einsatz bereit, mit denen Domino Integration auf höchster Ebene vorgenommen werden kann.

Domino beinhaltet verschiedene Funktionen und Werkzeuge, die wir in Verbindung mit XML nutzen. Neben den in den verschiedenen Programmierbereichen von Domino eingebundenen XML Funktionen, sind indem Domino Designer drei Teilbereiche vorhanden, mit denen bestimmte XML Funktionen ausgeführt werden können. Die Bereiche XML Betrachter, Exporter und Transformer können in Verbindung mit einem Standardbrowser die XML Inhalte von Domino direkt umsetzen und zur Anzeige bringen.

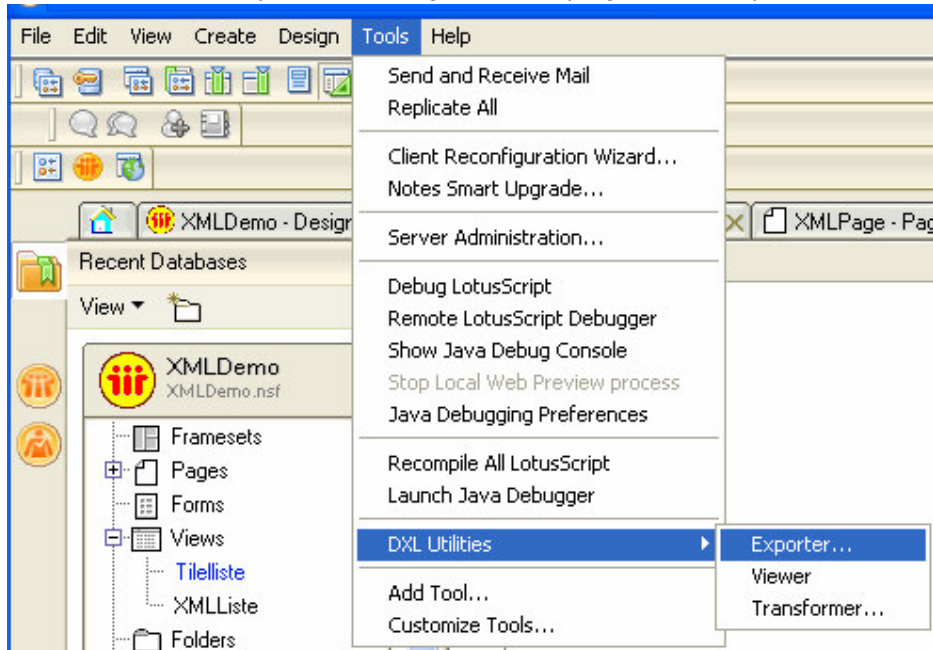
Der Betrachter dient dabei zum Beispiel der Anzeige der Gestaltungselemente einer Notes Datenbank in einem Browser.

Um diese Informationen darzustellen, wählen Sie zum Beispiel die gewünschte Ansicht und selektieren anschließend die Menüoption „Werkzeuge / DXL Hilfsprogramme / Betrachter“. Damit wird die Gestaltung des selektierten Bereiches in einem Browser in XML Format dargestellt. Ein Beispiel dafür finden Sie in Form der Darstellung der Kalenderansicht der Notes Datenbank, welche in einem Browser in XML Format ausgegeben wird:

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE view (View Source for full doctype...)
- <view name="Kalender" alias="Kalender" xmlns="http://www.lotus.com/dxl" version="8.0"
  replicaid="C125741F0028654D" showinmenu="true" noreplace="true" publicaccess="true"
  designerversion="8" language="en" unreadmarks="standard" onopengoto="lastopened"
  onrefresh="displayindicator" headers="simple" opencollapsed="false" showresponsehierarchy="false"
  showmargin="false" shrinkrows="true" extendlastcolumn="true" unreadcolor="#6260a1" rowlinecount="9"
  headerlinecount="1" rowspacing="1" bgcolor="white" totalscolor="gray" headerbgcolor="white"
  boldunreadrows="false" evaluateactions="true" allownewdocuments="true" allowcustomizations="true"
  hidemarginborder="false" marginwidth="0px" marginbgcolor="white" uniquekeys="false" useapplet="false"
  treatashtml="false" default="false" private="false" defaultdesign="false" allowdocselection="false"
  colorizeicons="false" direction="lefttoright" indexrefresh="autofirstuse" indexdiscard="inactive45days"
  noviewformat="false" type="standard">
- <noteinfo noteid="1cae" unid="A5232B7CBC9C0534C1257420005F826B" sequence="2">
- <created>
  <datetime dst="true">20080403T192313,07+02</datetime>
</created>
- <modified>
  <datetime dst="true">20080403T192330,86+02</datetime>
</modified>
</view>
```

Die Exporter Definitionen verfügen über Funktionen, mittels deren die Domino Gestaltungsbereiche in XML exportiert werden können, um diese in anderen Umgebungen einzusetzen. Um solche einen Export durchzuführen, führt man lediglich die folgenden Teilschritte aus:

1. Im Domino Designer wählen Sie die gewünschte Datenbank aus:
2. Selektieren Sie die Ansicht in der Auflistung der verfügbaren Ansichten.
3. Wählen Sie die Menüoption „Werkzeuge/DXL-Hilfsprogramme/Exporter“.



4. In dem sich öffnenden Fenster geben Sie nun den Namen und den Speicherort der XML Datei an und bestätigen Sie diese Angaben mit einem Klick auf „OK“.

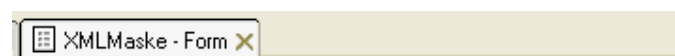
Basierend auf dieser XML Exportangabe der Ansichtsbeschreibung kann nun beispielsweise eine XML Definition erfolgen, welche in Form von XSL später als Vorgabe oder Schablone genutzt werden kann, um die Anzeige der XML Daten in Notes zu automatisieren.

Schauen wir und nun einmal an, wie man XML in Notes so nutzen kann, dass man XML Elemente in die Datenbank einbindet um damit eine Basis für einen Austausch zwischen Domino und Nicht-Domino Datenbanken realisieren zu können.

XML lässt sich, angelehnt an die Programmierung bzw. Codierung von HTML in Form von Tags angeben. Diese können bei Bedarf in den Code von beispielsweise einer Maske eingebunden werden.

Damit lassen sich Felder einzeln oder in Kombination aus einer Maske in XML Definitionen einbinden, die dann wiederum mit XML Werkzeugen weiter verarbeitet werden können.

In der nachfolgenden Abbildung finden Sie eine Maske mit integrierten XML Komponenten.



Dies ist ein Beispiel einer Maske mit XML Integration

```
<?xml version = "1.0"?>  
  
<teil>  
  <artikel> Artikel </artikel>  
  <menge> Menge </menge>  
  
</teil>
```

In dieser Abbildung sehen Sie ein einfaches Beispiel für die Integration von XML in einer Maske des Domino Designers. Dort ist ein XML Konstrukt dargestellt, welcher für einen Teilestamm genutzt werden kann, der nicht nur in Domino Anwendung findet, sondern zusätzlich vielleicht auch an die DB2/UDB des IBM i weitergegeben werden soll.

Schauen wir uns diese Maske einmal etwas genauer an:

Neben den bekannten Felddefinitionen „Artikel“ und „Menge“, mit denen die Felder beschrieben werden, haben wir in der Maske die typischen XML Komponenten integriert, welche die eigentlichen Domino Daten in dem Dokument bzw. der Maske auch als extern verwendbare Daten erscheinen lassen.

In dem Bereich „teil“, welcher als Gruppenbereich definiert wurde, sind die beiden Felder als Art Unterfelder zusammen gefasst.

Ich möchte an dieser Stelle nicht in die Details von XML eingehen, sondern versuchen, die Domino Besonderheiten aufzuzeigen.

Eine Besonderheit ist bei der XML Integration zu beachten: Die XML Daten werden in Abhängigkeit der Einstellungen von Domino mehr oder weniger korrekt dargestellt. Damit die in der vorhergehenden Abbildung enthaltenen Daten richtig ausgegeben werden, muss in den Maskenvorgaben der Inhaltstyp „HTML“ voreingestellt werden.

The screenshot shows the 'Form' properties dialog box. The 'On Web Access' section is expanded, and the 'Content type' is set to 'HTML', which is highlighted by a black arrow. Other options in this section include 'Formulas inherit values from selected document', 'Inherit entire selected document into rich text field', 'Automatically enable Edit Mode', 'Show context pane', 'Present mail send dialog', 'Character set' set to 'Default', 'Generate HTML for all fields', and color pickers for 'Active link' (red), 'Unvisited link' (blue), and 'Visited link' (purple). The bottom section contains 'Data Source Options' with fields for 'Default data connection' and 'Default metadata object'.

Diese Einstellvorgaben kennen wir bereits von Situationen, bei denen eine Maske im Wesentlichen als Webseite zum Einsatz kommen sollte. Wenn man nun aber auch XML Bestandteile in einer Maske vorfindet, dann sollte man neben den HTML Anzeigekomponenten auch dafür sorgen, dass die XML Bereiche korrekt von dem Webserver bzw. den Anwendungen umgesetzt werden können. Deshalb ist noch ein weiterer Schritt erforderlich, um die XML Darstellung zu gewährleisten:

Wählen Sie in den Eigenschaften der Maske in dem Info-Tabellenbereich die Option „in Notes als HTML rendern“.

Form

Name: XMLMaske

Comment: XML Maske

Type: Document

Display: ☒ Include in menu Create menu ☒ Include in Search Builder ☐ Include in Print

Versions: Versioning: None Create versions

Options: ☐ Default database form ☐ Store form in document ☐ Disable Field Exchange ☐ Automatically refresh fields ☐ Anonymous Form ☐ No Initial Focus ☐ No Focus On F6 ☐ Sign Documents that Use This Form ☒ Render pass through HTML in Notes ☐ Do not add field names to field index ☐ Allow Autosave

Conflict Handling: Create Conflicts

Mittels dieser Einstellungen lassen sich die XML Komponenten in einer Notes Maske einbinden und entsprechend verarbeiten.

Ähnliche Schritte sind bei der Integration von XML in Ansichten erforderlich. Damit XML in einer Ansicht eingebunden werden kann, müssen zunächst einmal die TAG-Anweisungen in den Ansichtscod aufgenommen werden. Dieses können wir zum Beispiel über eine Spaltenformel durchführen. Schauen Sie sich dazu das folgende Beispiel an, mit dem eine Artikelaufstellung in XML erzeugt wird:

„<teil><artikel>“+Artikel+“</teil><menge>“+menge+“</menge></teil>“

Mit diesem Codebeispiel, das wir in einer Spalte eingeben, listen wir die Artikeldaten in der Form auf, dass die Artikelnummer und der Lagerbestand in der Ansicht ausgegeben werden.

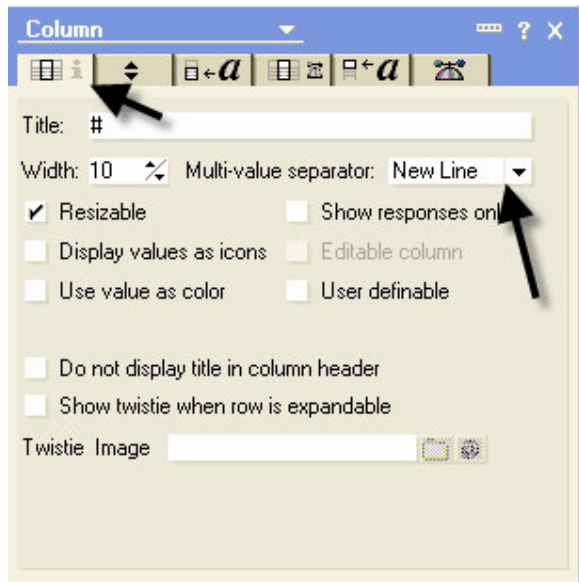
Geben Sie diesen Code in der Spaltenformel der Ansicht an:

(Column) : Column Value

Display ☐ Simple Function ☐ Field ☒ Formula

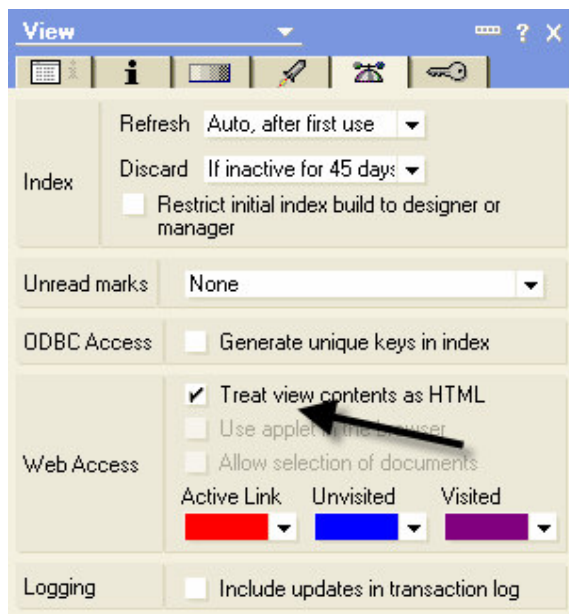
"<teil><artikel>“+Artikel+“</teil><menge>“+menge+“</menge></teil>“

Nun ist es noch wichtig, dass die Spalteneigenschaften dahingehend angepasst werden, dass die XML Daten korrekt dargestellt werden.



Wählen Sie dazu in den Ansichteigenschaften die Einstellung „Trennzeichen bei Mehrfachwert: Neue Zeile“, welche Sie in dem Info-Tabellenbereich finden. Schauen Sie sich dazu auch die vorhergehende Abbildung an.

Auch bei dem Beispiel „Ansicht“ müssen die HTML Eigenschaften mitgeteilt werden, damit später die Darstellung korrekt umgesetzt wird. Dazu definieren wir in den Eigenschaften der Ansicht in dem Tabellenbereich der erweiterten Optionen den Eintrag „Ansichtsinhalt als HTML behandeln“, wie es die nachfolgende Abbildung zeigt.



Damit die XML Bereiche korrekt dargestellt werden, müssen wir noch einen Seitenbereich definieren, der dazu dient, die XML Anfangs- und Endanweisungen anzugeben. Ein

Beispiel für eine solche Seite ist dieses:

```
<?xml version „1.0“ encoding =“utf-8“ ?>
```

```
<artikelliste>
```

```
TEIL
```

```
<teil><artikel>Schachbrett</artikel><menge>10</menge></artikel></teil>
```

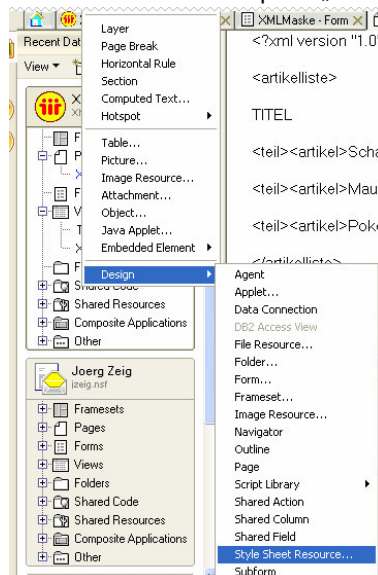
```
<teil><artikel>Mau Mau</artikel><menge>50</menge></artikel></teil>
```

```
<teil><artikel>Poker</artikel><menge>7</menge></artikel></teil>
```

```
</artikelliste>
```

Damit sind die Vorgaben für die XML Integration fast abgeschlossen. Was nun noch fehlt, ist eine Aufbereitungsangabe, wie die XML Daten bzw. Informationen in der Seite darzustellen sind. Diese ist zwar eigentlich optional zu sehen, sollte aber im Regelfall angewendet werden. Mittels eines XSL Stylesheets lassen sich die in dem XML Dokument enthaltenen Informationen aufbereiten. In der XSL Beschreibung befinden sich die Angaben, wie die XML- Inhalte darzustellen sind. Mit Hilfe dieser Schablonen können wir also das Erscheinungsbild der XML Inhalte direkt beeinflussen. Natürlich muss die XSL, welche genutzt werden soll, zuvor erstellt werden. In diesem Beispiel gehen wir davon aus, dass die XSL bereits existiert. Basierend auf dieser Beschreibung erstellen wir die notwendigen Einstellungen in dem Domino Designbereich. Die dazu erforderlichen Schritte lassen sich mit den im Domino Designer enthaltenen Werkzeugen ausführen:

1. Wählen Sie die Menüoption „Erstellen / Gestaltung / Stylesheet Ressource“.



2. Importieren Sie die gewünschte XSL Datei.
3. Passen Sie diese Datei so an, dass der folgende Zusatz darin aufgenommen wird:
`<?xml – stylesheet type="text/XSL" href="transform.xml"?>`
4. Speichern Sie diese Informationen.

Wenn die Ansicht mit diesen Einstellungen aufgerufen wird, dann erscheint dort die Auflistung der Artikel nach den in der XSL Definition enthaltenen Vorgaben.

Diese kurze Einführung in XML und Domino gibt einen kleinen Einblick in die Möglichkeiten der Verbindung dieser beiden Welten.