



# IBM Lotus Domino 8.5: Application Development is Reborn

---

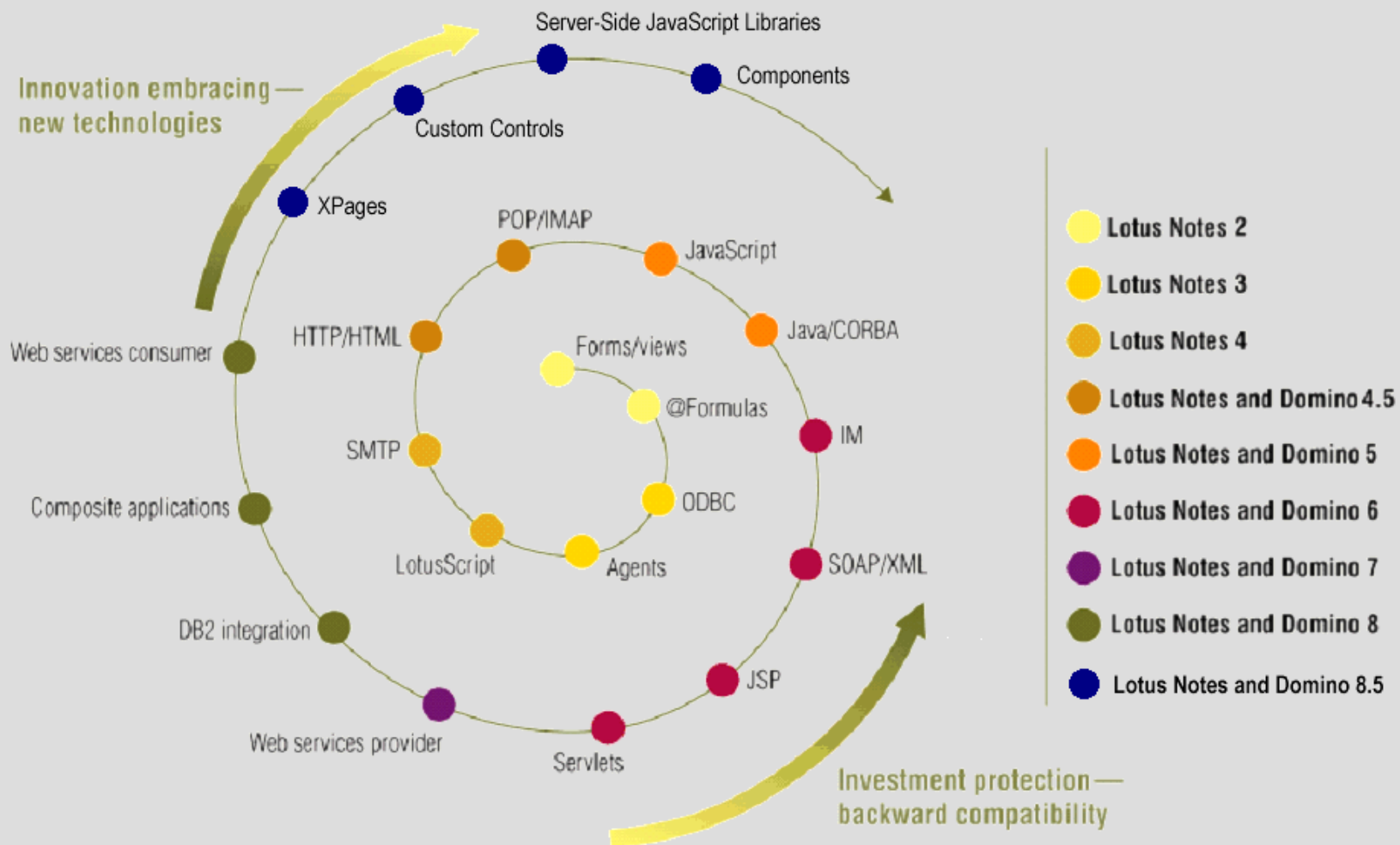
Maureen Leland  
Domino Designer Lead Architect, STSM  
IBM, Westford, MA

# Agenda

- Application Development in 8.5
- Designer 8.5
- Web Development in 8.5
- Beyond 8.5
- Q&A

# Domino Application Development

- Applications are at the heart of Domino success



## 8.5 Themes

- Provide developers with a modern development tool (IDE)
  - Do for developers what Notes 8 did for end users
    - Improve usability and productivity
    - Address developer pain points from previous releases
  - Attract new developers to the platform
- The Web has never looked better
  - Allow developers to build attractive, modern web apps that leverage Web 2.0 technologies with less work
  - Provide a means to incrementally update existing applications

# Agenda

- Application Development in 8.5
- Designer 8.5
- Web Development in 8.5
- Beyond 8.5
- Q&A

# Revolutionizing Domino Designer

- Same Designer productivity – built on Eclipse
  - Enhanced navigation and search capabilities
  - Increased customization
- Editing resources from within Designer
  - Web: HTML, XML,
  - JavaScript editors
  - Style sheets
- An extensible application development environment
  - Tools Plug in and integrate together

# Why Eclipse?

- Extensible script & debugger framework
- Built-in editors for CSS, XML, HTML, etc.
- Leverage continuing improvements in Eclipse “for free”
- Framework maintenance reduced, allows us to focus on your feature requests
- Allows you to work with multiple tools in a single container
- Partners can extend the tool using extension points

# Why Eclipse?



Let's See!

# Agenda

- Application Development in 8.5
- Designer 8.5
- Web Development in 8.5
- Beyond 8.5
- Q&A



# From Cool Tools to Cool Pages to XPages!

- Visual Web page construction
  - Advanced U/I controls
  - Drag and drop development
- Instant Application Development
  - XML markup with source
  - Full control of HTML
  - Presentation using CSS
  - Business logic using JavaScript
- Leverages existing NSF assets
  - Web agents, formulas, security,...

The screenshot displays the IBM Domino Designer interface. On the left, the 'Project Explorer' shows a project named 'SiteFinderDemo.nsf' with a tree structure including 'Forms', 'Views', 'Folders', 'XPages', 'Custom Controls', 'Framesets', 'Pages', 'Shared Elements', 'Code', 'Data', 'Resources', 'Composite Applications', and 'Application Properties'. The 'XPages' folder is expanded to show 'Site1', 'SiteFinal', 'SiteFinderFinal', 'SiteList1', and 'SiteListFinal'. The 'SiteFinderFinal' XPage is selected. Below the project explorer, the 'Tool Rentals' section shows 'toolmgr.nsf'. The 'Outline' panel at the bottom left shows a view structure with 'View: viewPanel', 'xp:this.data', 'View column', and 'View column'. The main design view shows a search page titled 'renovations' with a 'Site Finder' section. It includes a search form with fields for 'City', 'State', 'Country', and 'Zip/Postal code', a 'Search' button, and a 'Query string: {computedField1}'. Below the search form is a table with columns 'Site', 'Phone', 'Address', 'City', and 'State/Province'. The 'Properties' panel at the bottom right shows the 'Data' properties for the selected view, including 'Domino View: SiteList', 'View Column: Phone', and 'Display type: Number'.

# What are XPages?

- New design element fully integrated within an NSF/NTF
- Defines a page that can be rendered in a browser
- Served up by the Domino web server
- Native, fast access to the data in your NSF
- Allows you to build beautiful, dynamic, well performing pages
- Mixes and matches with other design elements in your application
- Easily localized for global applications
- Built on Java™ Server Faces, so architected to support more than just browser platforms (eventually)



# Do I Have to Rewrite My Application?

---

**NO!!!!**



# Do I Have to Learn Java and JSF?

---

**NO!!!!**



# Can I Call Java if I Want To?

---

**YES!!!**

# Data on Your XPage

- XPages separate UI from data
  - clarity in application design
  - freedom of presentation
- Natural integration with Domino views and documents, future integration with other sources of data and services
- Multiple sources of data can be collected on a single page
  - Data can be from another NSF
- Improved web rich text handling
  - Preserves embedded images, better handling of sections/tables
  - If page submitted, rich text field updated only if data changed
  - Treated as MIME, MIME rendering improved

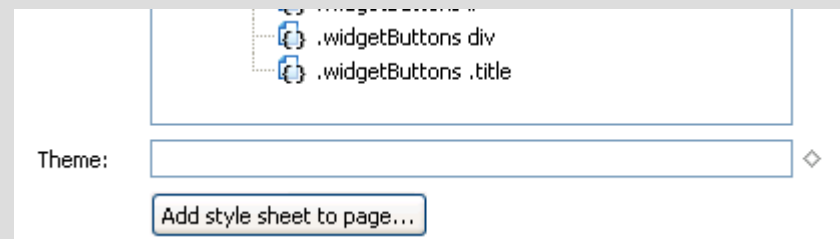
# Logic on Your XPage

- The “official” language for XPages is JavaScript
- JavaScript runs on either the client or on the server (or both!)
- Server JavaScript includes your familiar @functions (reimplemented for XPages)
  - @UserName, @DbLookup, @DbColumn, etc.!!
- Simple actions available for common operations
- (Almost) every property is computable!
  - Labels, column titles, visibility, style, ....
- Ajax behaviors
  - builtin in some cases (typeahead, pager)
  - easy to build on your own with simple event properties

# Themes: Skinning and More

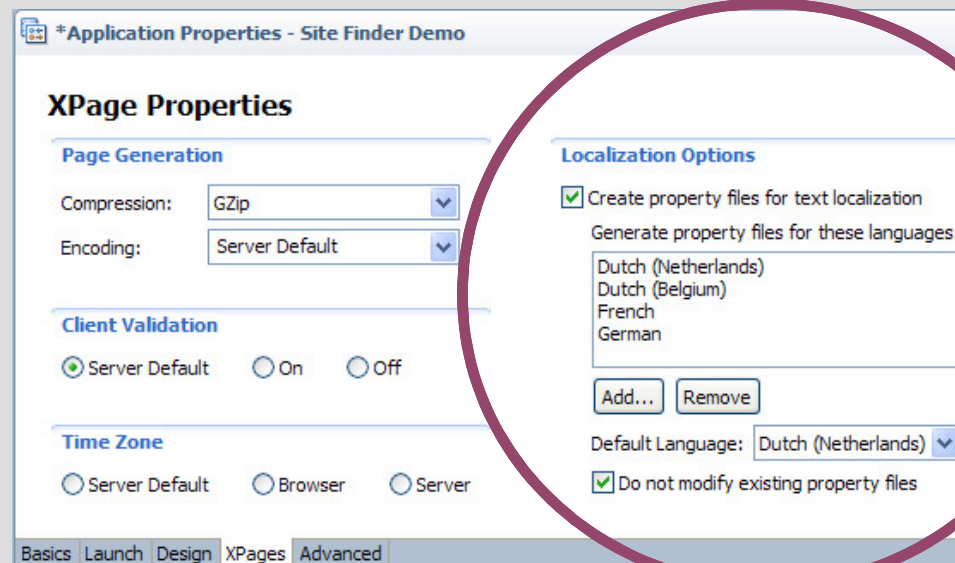
- Allows an application to pick up a look and feel without touching the actual application code
  - Default styles are applied to controls without any explicit style set per control – every property can be set
  - Skinning can be disabled per application/control
  - Association between control done through a “Theme”
  - Defined at different levels: server or application
    - Stored in a shared database
  - Easily customizable by a developer and/or an administrator

```
<theme>
  <resource>
    <content-type>text/css</content-type>
    <href>/_ibmxspres/global/theme/webstandard/xsp.css</href>
  </resource>
  <!-- Global controls -->
  <!-- View Root (Page Body) -->
  <control>
    <name>ViewRoot</name>
    <property>
      <name>styleClass</name>
      <value>xspView</value>
    </property>
  </control>
```



# Localization

- XPages are fully localizable
- Automated string extraction process that generates resource bundles
- No runtime penalty for localized pages

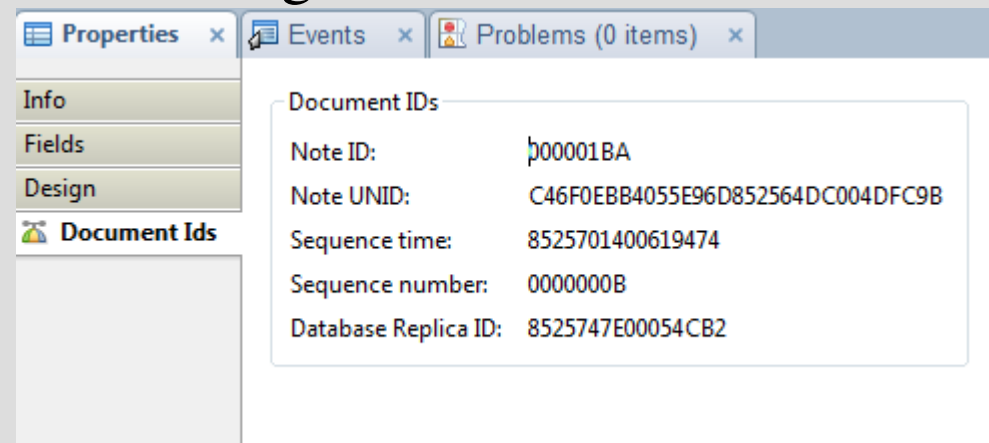


# Agenda

- Application Development in 8.5
- Designer 8.5
- Web Development in 8.5
- Beyond 8.5
- Q&A

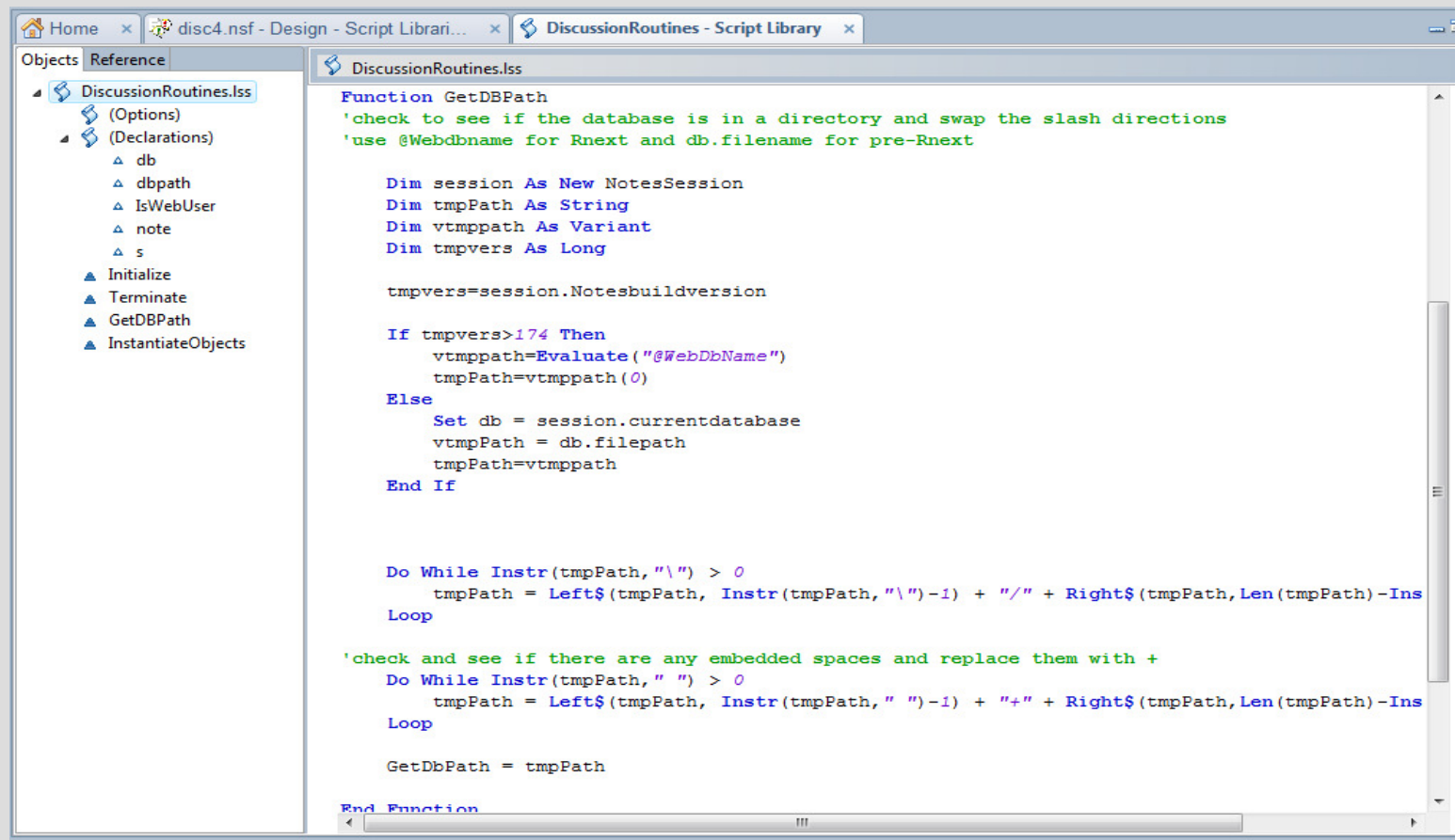
# What's Next? Designer 8.5.1

- Focus on first things first....
- Performance improvements
- Extension API
- Usability improvements
  - New XPages dialog
  - Working sets
  - Finishing touches – some little things...



# And Some Larger Finishing Touches

- A new Eclipse-based LotusScript editor



```
Function GetDBPath
'check to see if the database is in a directory and swap the slash directions
'use @Webdbname for Rnext and db.filename for pre-Rnext

Dim session As New NotesSession
Dim tmpPath As String
Dim vtmppath As Variant
Dim tmpvers As Long

tmpvers=session.Notesbuildversion

If tmpvers>174 Then
    vtmppath=Evaluate("@WebDbName")
    tmpPath=vtmppath(0)
Else
    Set db = session.currentdatabase
    vtmpPath = db.filepath
    tmpPath=vtmppath
End If

Do While Instr(tmpPath,"\" ) > 0
    tmpPath = Left$(tmpPath, Instr(tmpPath,"\" )-1) + "/" + Right$(tmpPath, Len(tmpPath)-Insr
Loop

'check and see if there are any embedded spaces and replace them with +
Do While Instr(tmpPath," ") > 0
    tmpPath = Left$(tmpPath, Instr(tmpPath," ")-1) + "+" + Right$(tmpPath, Len(tmpPath)-Insr
Loop

GetDbPath = tmpPath

End Function
```



# And Eclipse-based Java Editor

The screenshot displays the Eclipse IDE interface. On the left, the 'Java Library Contents' view shows a project named 'ExcelLibrary' with a source folder 'Src' containing 'ExcelDemo.java'. Below this, the 'Properties' view shows the 'Basics' tab with the name 'ExcelLibrary' and base class 'ExcelDemo.class'. The main editor area is titled 'Editing the Java Library' and shows the source code for 'ExcelDemo.java'. The code includes an import for 'java.io.FileInputStream', a class declaration 'ExcelDemo', and three methods: a constructor, 'showSheetNames()', and 'showDataForSheet()'. The code is as follows:

```
import java.io.FileInputStream;

public class ExcelDemo {

    private String fileName;
    private HSSFWorkbook workbook;

    public ExcelDemo(String fileName){
        this.fileName = fileName;

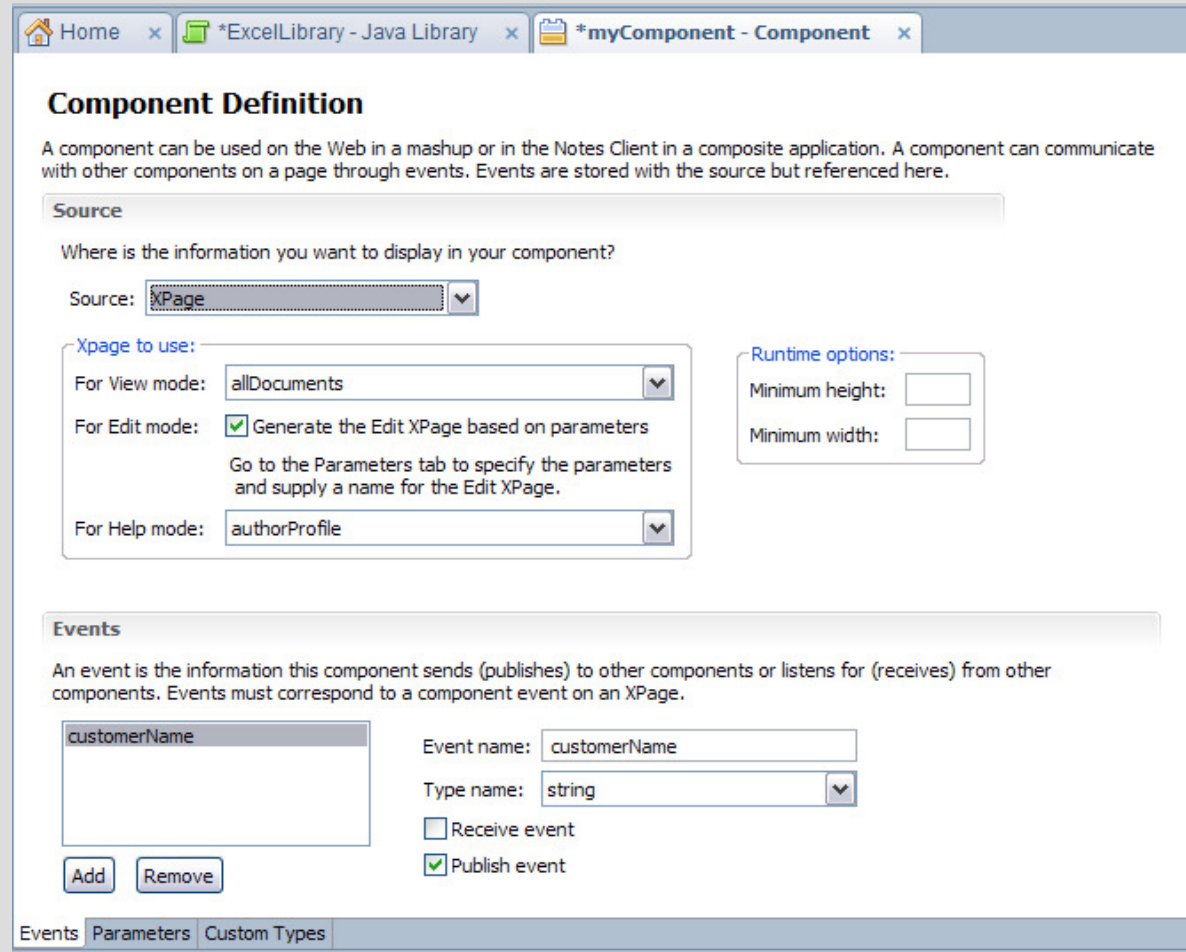
        try {
            POIFSFileSystem fs = new POIFSFileSystem(new FileInputStream(fileName));
            workbook = new HSSFWorkbook(fs);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public void showSheetNames(){
        for (int i = 0; i < workbook.getNumberOfSheets(); i++){
            System.out.println("Sheet Name :: " + workbook.getSheetName(i));
        }
    }

    public void showDataForSheet(int index){
        System.out.print("\nShowing Data of Sheet:: " + workbook.getSheetName(index));
        HSSFSheet sheet = workbook.getSheetAt(index);
```

# Domino Web Application Mashups!

- With the help of a new Component design element
- Wraps an XPage and defines any input properties, output events, or widget parameters



The screenshot shows the 'Component Definition' dialog box in Domino Designer. The 'Source' is set to 'XPage'. Under 'Xpage to use', 'For View mode' is 'allDocuments', 'For Edit mode' is checked 'Generate the Edit XPage based on parameters', and 'For Help mode' is 'authorProfile'. 'Runtime options' for 'Minimum height' and 'Minimum width' are empty. The 'Events' section shows an event named 'customerName' of type 'string', with 'Publish event' checked and 'Receive event' unchecked. The 'Parameters' tab is selected at the bottom.

**Component Definition**

A component can be used on the Web in a mashup or in the Notes Client in a composite application. A component can communicate with other components on a page through events. Events are stored with the source but referenced here.

**Source**

Where is the information you want to display in your component?

Source:

**Xpage to use:**

For View mode:

For Edit mode:  Generate the Edit XPage based on parameters  
Go to the Parameters tab to specify the parameters and supply a name for the Edit XPage.

For Help mode:

**Runtime options:**

Minimum height:

Minimum width:

**Events**

An event is the information this component sends (publishes) to other components or listens for (receives) from other components. Events must correspond to a component event on an XPage.

Event name:

Type name:

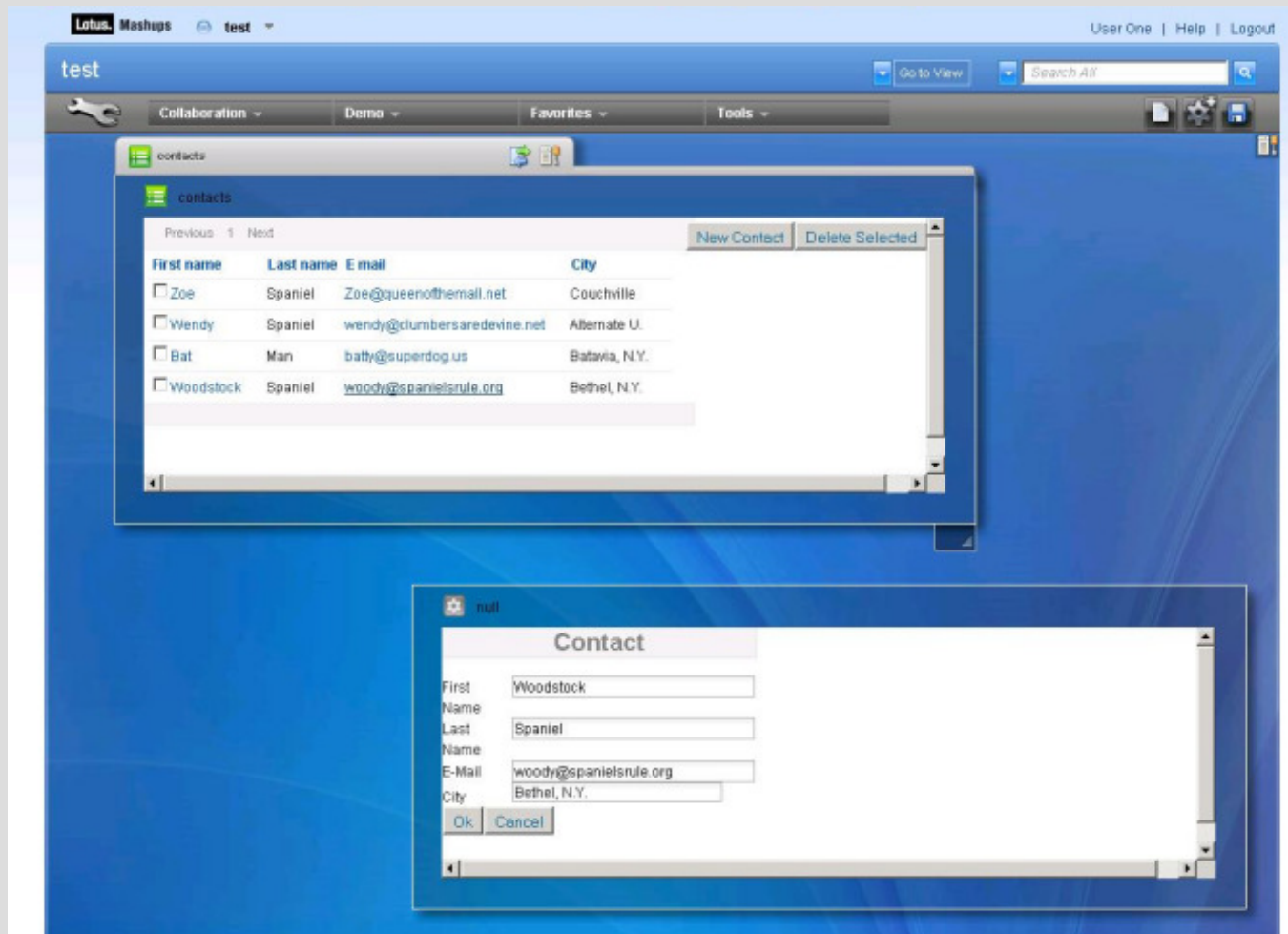
Receive event

Publish event

Events | Parameters | Custom Types

# Domino Mashup

- Widget is served up to Mashup Maker through the Domino web server



The screenshot displays a Domino Mashup Maker interface. The main window is titled 'test' and shows a 'contacts' widget. The widget contains a table with the following data:

First name	Last name	E mail	City
<input type="checkbox"/> Zoe	Spaniel	Zoe@queenofthemail.net	Couchville
<input type="checkbox"/> Wendy	Spaniel	wendy@clumbersaredevine.net	Altemate U.
<input type="checkbox"/> Bat	Man	batty@superdog.us	Batavia, N.Y.
<input type="checkbox"/> Woodstock	Spaniel	woody@spanielsrule.org	Bethel, N.Y.

Below the table is a 'Contact' form with the following fields:

First Name: Woodstock  
Last Name: Spaniel  
E-Mail: woody@spanielsrule.org  
City: Bethel, N.Y.

The form includes 'Ok' and 'Cancel' buttons.

# XPages in the Notes Client

- Full promise of XPages realized as it supports more platforms

# What Else?

- Support for more XPage data sources
- Debugging: Server JavaScript, LotusScript, ...
- More infoboxes on the move to property panels
- More platforms: Linux™, Mac OS®
- And....

# What Else?

- Support for more XPage data sources
- Debugging: Server JavaScript, LotusScript, ...
- More infoboxes on the move to property panels
- More platforms: Linux™, Mac OS®
- And....
- What Do You Want??

We Will Keep Listening!

# Summary



- Domino application development has had a revolution in 8.5
- The revolution is continuing in 8.5.1 and beyond!!!
- Designer and the applications it builds are brand new and more powerful than ever!

# Agenda

- Application Development in 8.5
- Designer 8.5
- Web Development in 8.5
- Beyond 8.5
- Q&A

# Resources



Lotus Domino Designer wiki  
[lotus.com/ldd/ddwiki.nsf](http://lotus.com/ldd/ddwiki.nsf)

Wiki articles

Videos

Red wikis

The screenshot shows the Lotus Domino Designer wiki homepage. At the top, there is a navigation bar with the Lotus software logo on the left and the IBM logo on the right. Below the navigation bar, the page title "Lotus Domino Designer wiki" is displayed. On the left side, there is a search box with the text "Search this wiki" and a "Go" button. Below the search box, there is a "Contribute" section with a list of links: "Log on", "Add an article", "Wiki feedback", "Wiki user name registration", "Wiki help", "Rules of conduct", and "Terms of use". Below the "Contribute" section, there is a "Navigation" section with a list of links: "Home", "Application design (32)", "Application management (2)", "Composite applications (13)", "Configuration (1)", "Databases (1)", "Demonstrations (1)", "FAQ (22)", "Getting started (7)", "Installation (0)", and "Lotus Connectors (0)".

The main content area features a "Welcome to the Domino Designer wiki" message. To the left of the message is a circular icon with a pencil and a document. To the right of the icon is a paragraph of text: "This community is for you to learn about IBM® Domino® Designer™, contribute to its knowledge base, and collaborate with others. It will help you create and deploy Domino applications and includes application design guidelines and best practices for developing web applications. We invite you to create new articles on these topics, or to expand our content to include samples and troubleshooting."

Below the welcome message, there is a section titled "About Domino Designer" with a paragraph of text: "With Domino Designer, IBM Lotus® Notes® and Domino provide a high productivity application development environment that supports the rapid creation, modification and deployment of security-rich business applications. Domino applications can be accessed using numerous methods, including Web service consumers, Lotus Notes, and Web browsers. Lotus Domino Designer 8 includes multiple features and functions that enable developers to provide more value through Lotus Notes and Domino applications and support services oriented architecture. These include the following:"

On the right side of the page, there is a "Latest wiki articles" section with a list of links: "Create an empty NotesDocumentCollection - November 26 2008", "Actual/uncached reading of profile document fields with lotusscript - November 21 2008", "New Editors in Domino Designer 8.5 - November 19 2008", and "Updating comboBox values on Xpage - November 18 2008".



# Legal Disclaimer

© IBM Corporation 2009. All Rights Reserved.

The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.

IBM, the IBM logo, Lotus, Lotus Notes, Notes, Domino, Quickr, Sametime, WebSphere, UC2, PartnerWorld and Lotusphere are trademarks of International Business Machines Corporation in the United States, other countries, or both. Unyte is a trademark of WebDialogs, Inc., in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.