



# **IBM Domino® Designer Tips, Tricks and Enhancements for Maximum Productivity!**

Gary Marjoram, IBM

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Please Note

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

# Speaker Introductions



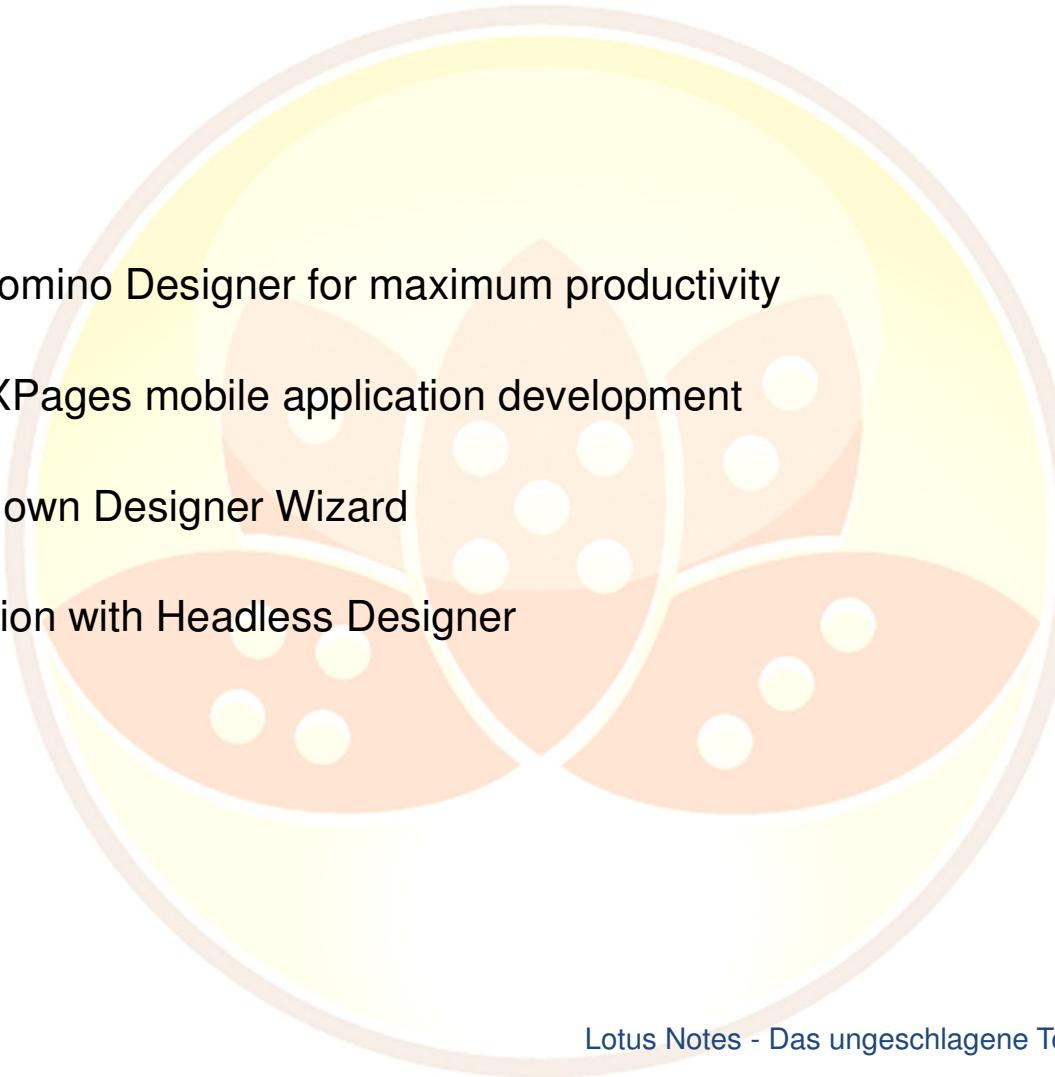
## Gary Marjoram - Software Engineer : IBM Ireland

- Joined the IBM Domino® Designer team and IBM in April 2013
- Over 18 years experience in the Software Industry across multiple industries including telecoms, automotive and pharmaceutical
- @gmarjoram\_ibm
- marjoram@ie.ibm.com

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Agenda

- Configuring Domino Designer for maximum productivity
- Accelerating XPages mobile application development
- Creating your own Designer Wizard
- Build Automation with Headless Designer





## Configuring Domino Designer for maximum productivity

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# JVM Performance

Do this today!

- Domino Designer/Notes® is a Java program
- It requires a JRE to run
- The JVM has low memory settings in order to keep the Notes footprint as small as possible
- Designer requires more memory due to the large number of editors and builders it supports
- **Designer performance is VASTLY improved with these recommended settings:**

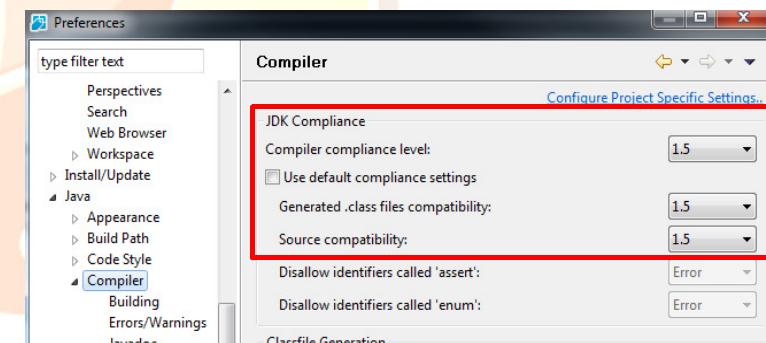
Designer 9.0.1 Default Settings	Designer 9.0.1 Recommended Settings (Stand-alone Installation)
<code>vmarg.Xmx=-Xmx512m</code> <code>vmarg.Xms=-Xms48m</code> <code>vmarg.Xmca=-Xmca512k</code>	<code>vmarg.Xmx=-Xmx1024m</code> <code>vmarg.Xms=-Xms512m</code> <code>vmarg.Xmca=-Xmca512k</code>

- `jvm.properties` is located at `<notes_bin>\framework\rcp\deploy`
- Full details available in this technote:
  - <http://tinyurl.com/DesignerJVMSettings>

# Build Automatically

To build or not to build that is the question!

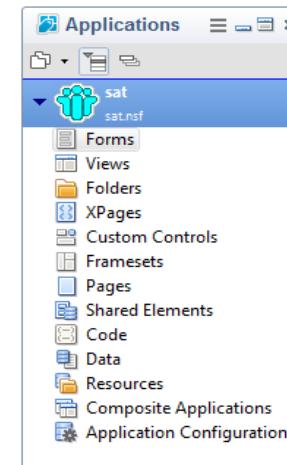
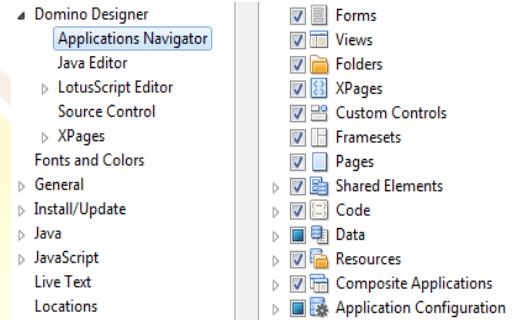
- Consider turning build automatically off if you have large applications on remote servers
  - Eclipse Java Builder is not optimized to execute over the network
- Mainly a matter of user experience
  - Novice users tend to forget to manually 'build' the application
- Java compile setting
  - Do not modify the Java->Compiler->JDK Compliance settings
- Domino remains backwards compatible
  - Java artifacts written on 9.0.1 will run on previous versions
  - As a result, the compliance is set to 1.5 automatically by Designer on startup



# Customize your environment

Reduce the clutter!

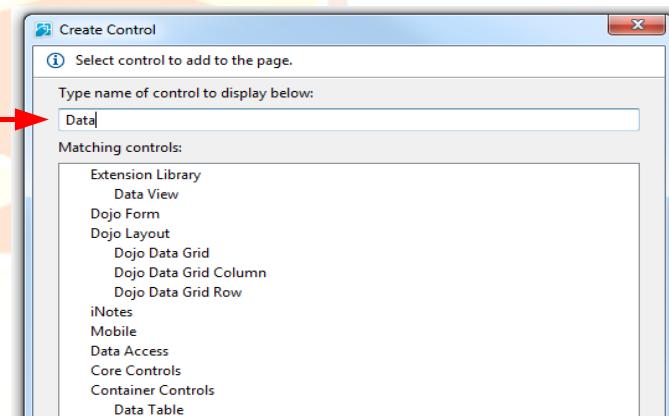
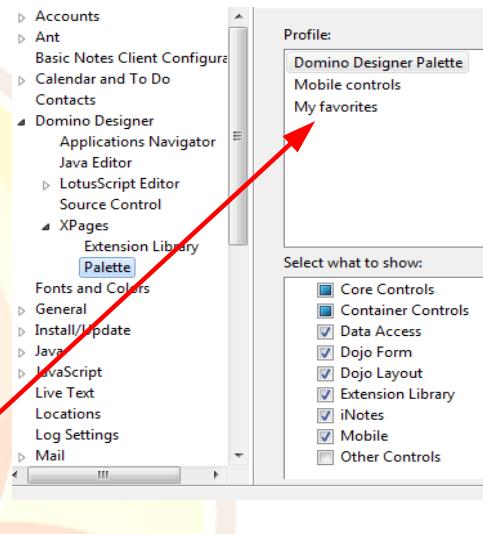
- Customize existing perspectives (Designer, XPages ...)
  - Discard unused Views (Panels)
  - Re-arrange Views (Panels)
- Create and save new perspectives
- Customize the “Applications Navigator” in preferences
  - Remove design elements you don't use
- Create working sets
  - Reduces clutter in the Navigator
  - Rapidly change focus



# Customize your environment

Reduce the clutter!

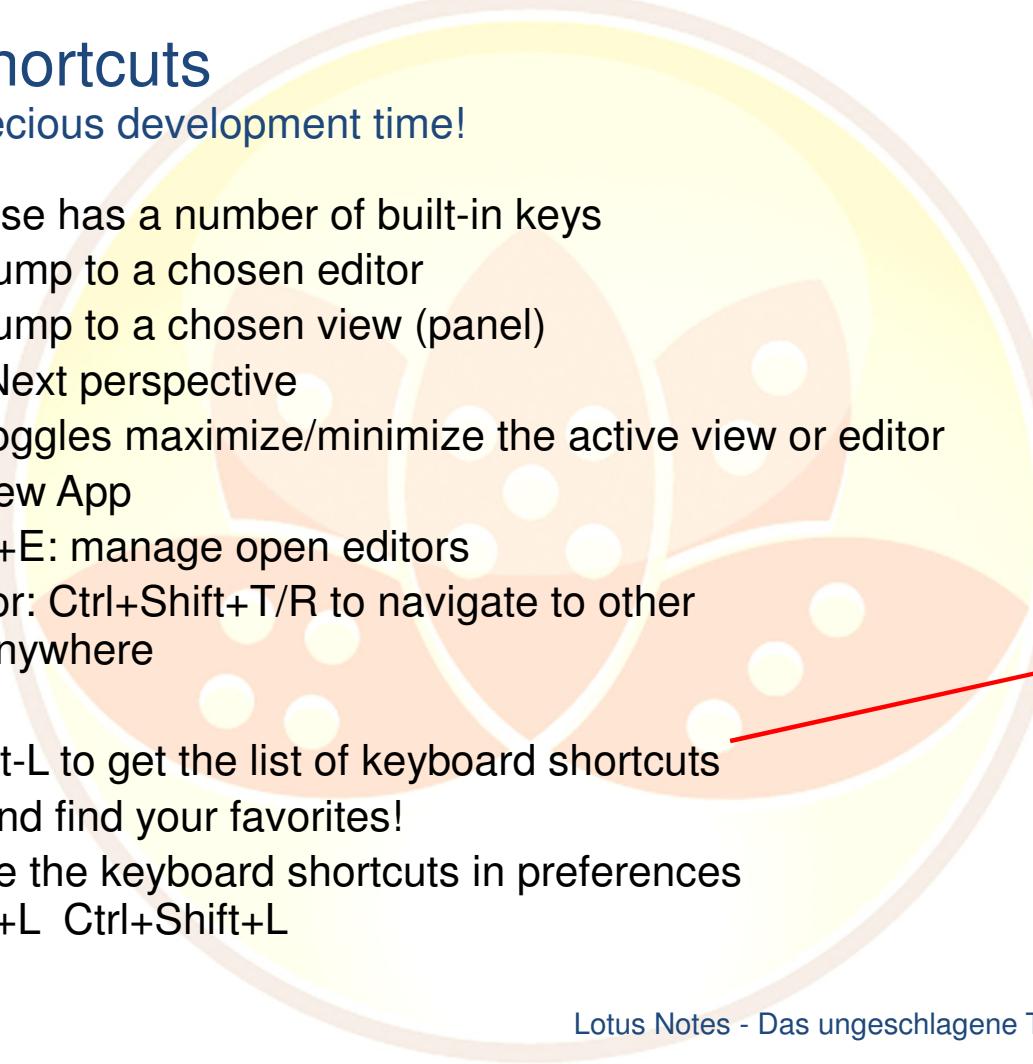
- Close Applications when not actively working on them
  - Frees valuable memory!
- For XPages development customize the Controls Palette in preferences
  - Remove unused / rarely used controls
  - Create custom palettes
  - Can't find a control? Did you know that you can search the palette?



# Keyboard Shortcuts

Shortcuts save precious development time!

- Designer/Eclipse has a number of built-in keys
  - Ctrl+F6: jump to a chosen editor
  - Ctrl+F7: jump to a chosen view (panel)
  - Ctrl+F8: Next perspective
  - Ctrl+M: Toggles maximize/minimize the active view or editor
  - Ctrl+N: New App
  - Ctrl+Shift+E: manage open editors
  - Java editor: Ctrl+Shift+T/R to navigate to other classes/anywhere
- Press Ctrl-Shift-L to get the list of keyboard shortcuts
  - Explore and find your favorites!
  - Customize the keyboard shortcuts in preferences  
Ctrl+Shift+L



A red arrow points from the text "Press Ctrl-Shift-L to get the list of keyboard shortcuts" towards this table.

Activate Editor	F12
Add Javadoc Comment	Alt+Shift+J
All Instances	Ctrl+Shift+N
Backward History	Alt+Left
Build All	Ctrl+B
Change Method Signature	Alt+Shift+C
Close	Esc
Close All	Ctrl+Shift+W
Collapse All	Ctrl+Shift+Numpad_Divide
Content Assist	Ctrl+Space
Context Information	Ctrl+Shift+Space
Copy	Ctrl+C
Cut	Ctrl+X
Debug Ant Build	Alt+Shift+D, Q
Debug Eclipse Application	Alt+Shift+D, E
Debug JUnit Plug-in Test	Alt+Shift+D, P
Debug JUnit Test	Alt+Shift+D, T
Debug Java Applet	Alt+Shift+D, A
Debug Java Application	Alt+Shift+D, J

Press "Ctrl+Shift+L" to open the preference page.

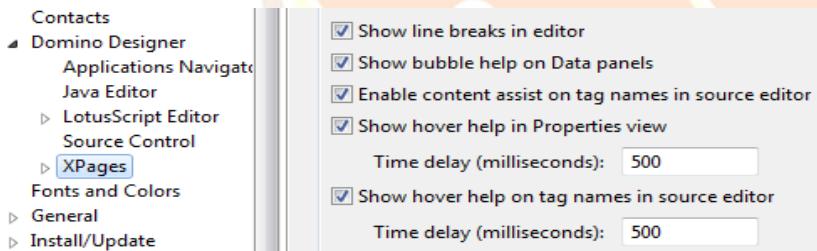
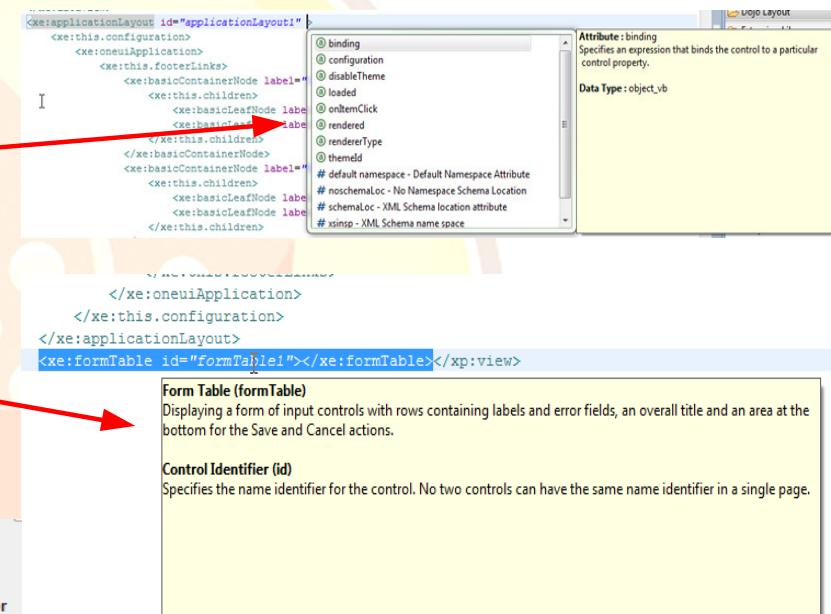
# Improving XPages Developer Productivity

## XSP Source Editor

- Writing XSP source code can be simplified using these aids

- Content assist <ctrl+space> (also works with SSJS)
- Hyperlink navigation
- Hover help

- Preferences exist to customize their behavior



# Improving XPages Developer Productivity

## XSP Snippets

- Commonly used XSP Snippets can be defined using the XML editor preferences
- Snippets can then be inserted to an XPage, in source mode, by hitting <ctrl+space> and typing a snippet name

The screenshot shows the XML Editor's preferences panel. Under the 'Editor' section, the 'Templates' option is selected, highlighted with a red arrow. To the right is a table titled 'Templates' with columns for Name, Context, Description, and Auto Ins... (set to 'on' for all). Below the table is a 'Preview' section displaying the XSP code for the 'keepSessionalive' template.

Name	Context	Description	Auto Ins...
comment	XML Tag	xml comment	on
debug	All XML		on
default namespace	XML Attribute	Default Namespace Attribute	on
dump	XML Tag		on
keepSessionalive	XML Tag		on
noschemaLoc	XML Attribute	No Namespace Schema Location	on
schemaLoc	XML Attribute	XML Schema location attribute	on
xml declaration	New XML	xml declaration	on
xsinsp	XML Attribute	XML Schema name space	on
XSL processing instru...	XML Tag	XSL processing instruction	on

Preview:

```
<xp:scriptBlock id="scriptBlock1">
<xp:this.value>
<![CDATA[
XSP.addOnLoad(function() {
    setInterval(function() {
        XSP.partialRefreshPost("#{id:keepSessionAlive}", {});
    }, 3000000
})]);
</xp:this.value>
</xp:scriptBlock>
```

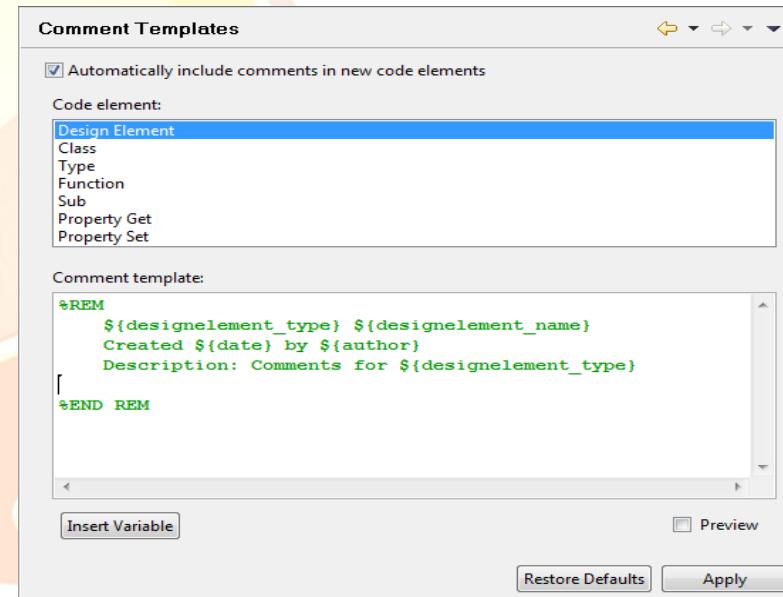
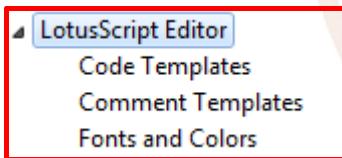
The screenshot shows an XPage source editor window. The code editor has a yellow background where the inserted XSP snippet is highlighted. The code is identical to the one shown in the XML Editor's preview.

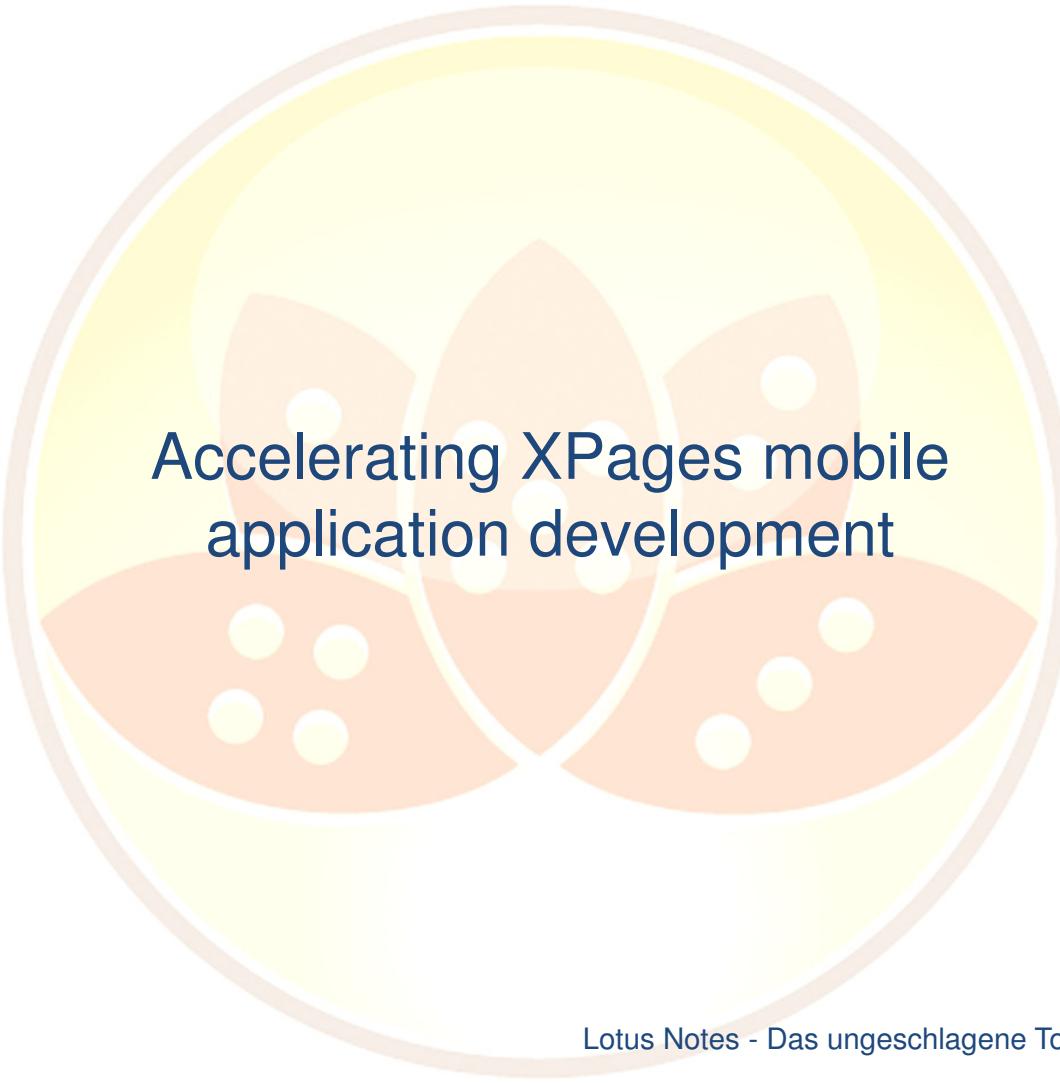
```
<xp:scriptBlock id="scriptBlock1">
<xp:this.value>
<![CDATA[
XSP.addOnLoad(function() {
    setInterval(function() {
        XSP.partialRefreshPost("#{id:keepSessionAlive}", {});
    }, 3000000
})]);
</xp:this.value>
</xp:scriptBlock>
```

# Improving Developer Productivity

## LotusScript Editor

- Code and Comment Templates
  - Allow users to automatically insert user defined code and comment templates on predefined artifacts/events (e.g. new Script Lib, new Sub etc..)
  - Allows organizations to implement certain coding standards easily (e.g. all Design elements have a copyright comment header etc)
- Font, colors etc..
  - Define which fonts (sizes and colors) are to be used for various language syntax





## Accelerating XPages mobile application development

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Domino Designer for Mobile App Dev

The *Single Page Application Wizard* - what is it?

- The Single Page Application Wizard is part of the “**com.ibm.xsp.extlib.designer.tooling**” plugin in the XPages Extension Library
- It is a Designer extension that contributes to the “**com.ibm.designer.domino.xsp.editor.component**” extension point
- It has been available on OpenNTF.org since 13<sup>th</sup> December 2013



XPages Extension Library - Release: 901v00\_02.20131212-1115

Release Name	901v00_02.20131212-1115	Released on	Dec 13, 2013
License	Apache License	Released by	Paul Hannan
Status	Released	Downloads	108
		Cleared	Yes

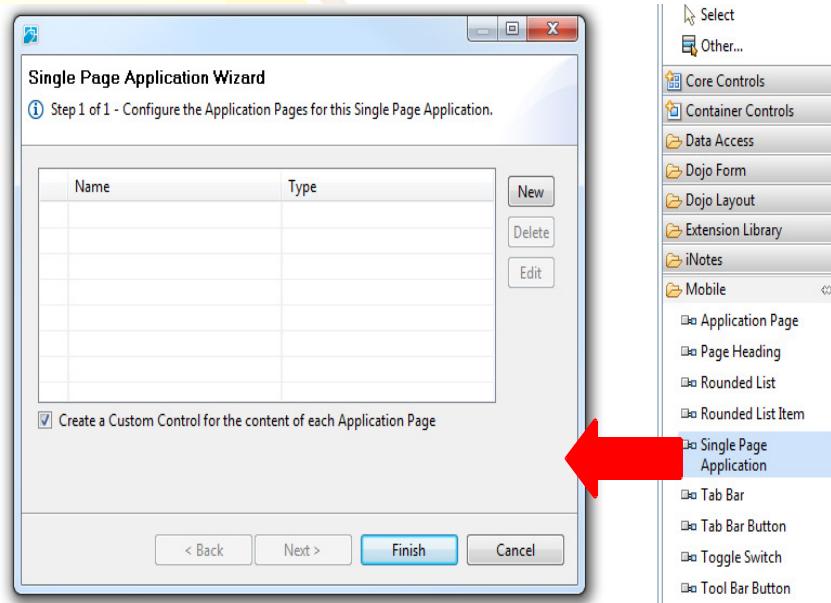
Download(s)

[ExtensionLibraryOpenNTF-901v00\\_02.20131212-1115.zip](#)

# Single Page Application Wizard

Rapidly design your XPages mobile application

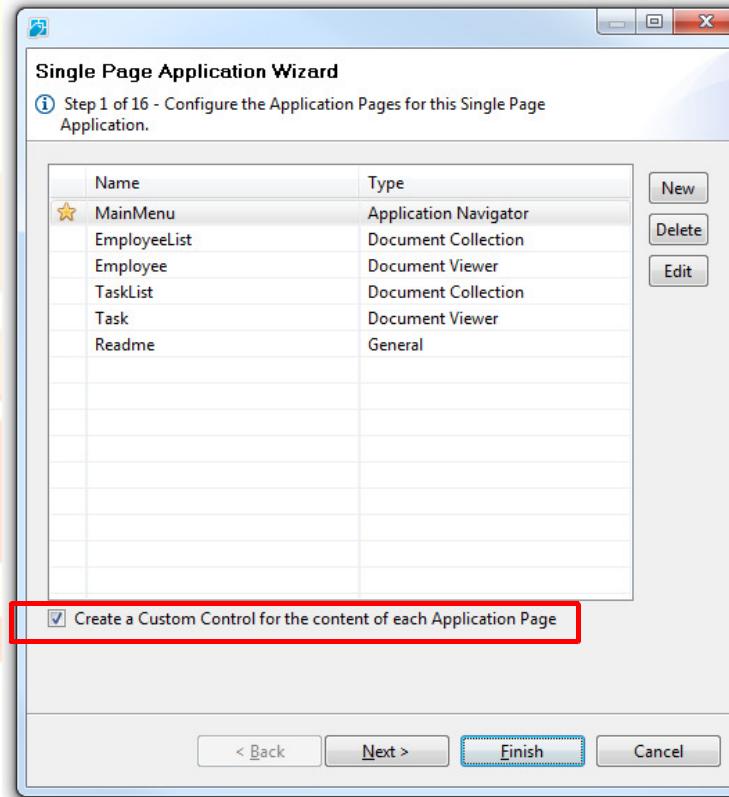
- The Wizard allows you to create an initial design framework for your XPages mobile application
- It walks you through the process of creating new Application Pages (xe:appPage) and populating them with various mobile controls
- Allows you to configure navigation between the Application Pages
- The produced App can perform basic CRUD with action buttons defined using the Wizard



# Single Page Application Wizard

## Adding Application Pages

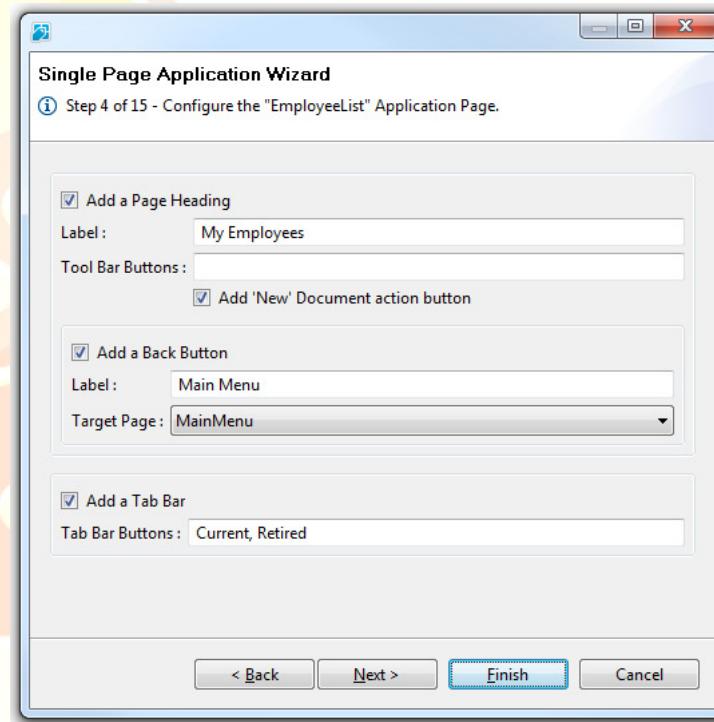
- Design your application by adding Application Pages and specifying their type
  - General
  - Navigator
  - Document Collection
  - Document Viewer
- Select this option to produce a Custom Control for the content of each Application Page
  - **This is the recommended design pattern for XPages mobile applications**



# Single Page Application Wizard

## Configure the main options for each Application Page

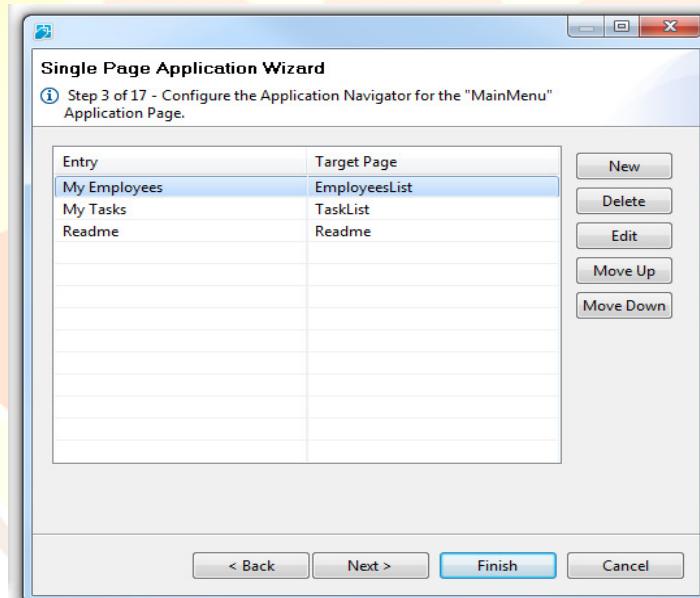
- Configure the Page Heading
  - Add a Page Label
  - Add a Tool Bar with buttons
  - Add a Back button and specify the Target Application Page
- Configure the Tab Bar with buttons
  - Buttons defined with a comma separated list
- Add predefined Document action buttons
  - For Document Collection and Viewer type pages action buttons can be added here to allow basic CRUD operations on the target application



# Single Page Application Wizard

Configure specific options for each page type

- Navigator
  - Add a list of links to other Application Pages

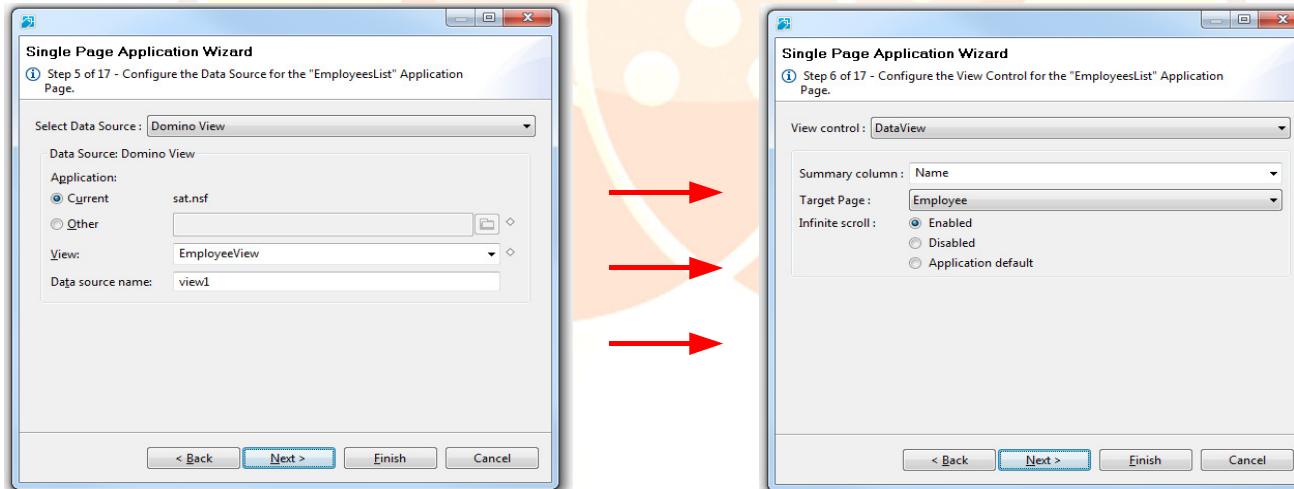


# Single Page Application Wizard

Configure specific options for each page type

- Document Collection

- Define a data source
- Choose the summary column for the Data View
- Specify the target Document Viewer
- Infinite scroll options



# Single Page Application Wizard

Configure specific options for each page type

- Document Viewer
  - Define a data source
  - Configure the labels, fields and controls for this document

The image shows two windows of the Single Page Application Wizard. The left window, titled 'Step 8 of 17 - Configure the Data Source for the "Employee" Application Page', contains fields for selecting a data source (Domino Document), application (Current: sat.nsf), form (EmployeeForm), and other configuration options like default action and run form validation. The right window, titled 'Step 9 of 16 - Configure the Fields for the "Employee" Application Page', lists fields (Name, Title, Dob, Desc) with their labels and controls (Edit Box, Edit Box, Date only, Rich Text). Red arrows point from the right side of the left window towards the right window, indicating a flow or relationship between the two steps.

**Step 8 of 17 - Configure the Data Source for the "Employee" Application Page.**

Select Data Source : Domino Document  
Data Source: Domino Document  
Application:  
     Current     sat.nsf  
     Other  
Form: EmployeeForm  
Data source name: document1  
Default action: Create document  
Parent id:  
(for responses only)  
Document id:  
Run form validation:

**Step 9 of 16 - Configure the Fields for the "Employee" Application Page**

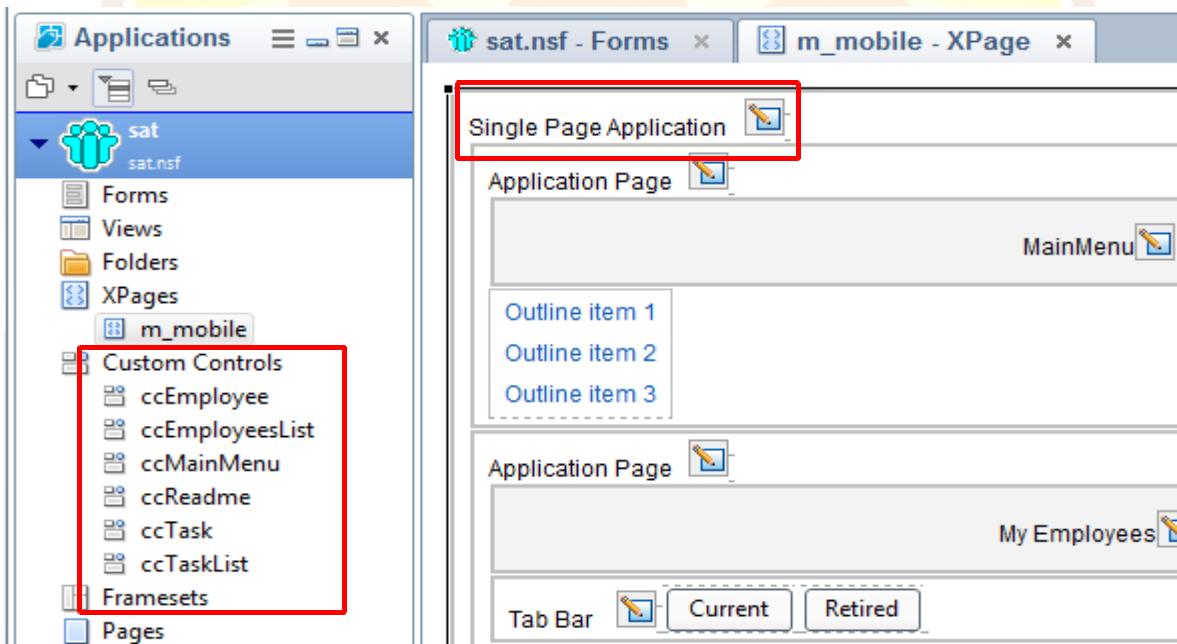
Field	Label	Control
<input checked="" type="checkbox"/> Name	Name:	Edit Box
<input checked="" type="checkbox"/> Title	Title:	Edit Box
<input checked="" type="checkbox"/> Dob	Date of birth:	Date only
<input checked="" type="checkbox"/> Desc	Description:	Rich Text

Reset  
Deselect All  
Edit  
Move Up  
Move Down

# Single Page Application Wizard

What does it produce?

- When Finish is clicked the configured Single Page Application control is added to the XPage and Custom Controls are created, if any



# Single Page Application Wizard

Preview your mobile application

- The Wizard outputs a fully functional mobile application, capable of CRUD, with the navigation between pages in place

The screenshot displays three mobile application screens arranged horizontally, connected by a central vertical navigation bar.

- Main Menu:** A black header with the text "Main Menu". Below it is a list of items: "My Employees", "My Tasks", and "Readme".
- My Employees:** A black header with the text "My Employees". It includes a "Main Menu" button, a "New" button, and two filter buttons: "Current" and "Retired". The main content area lists employee names: Bob Alexander, Fred O'Neill, Michael Murphy, and Brian McDonnell, each followed by a right-pointing arrow.
- Employee:** A black header with the text "Employee". It includes a "Employees" button. The main content area displays detailed information for Bob Alexander:
  - Name: Bob Alexander
  - Title: Programmer
  - Date of birth: Jul 12, 1973
  - Description: Team Lead

# Single Page Application Wizard

## In Summary ...

- Produces a fully-functional well-structured mobile app, capable of CRUD operations, without requiring expertise in mobile design, XPages mechanics or XSP markup specifics
- Implements Single Page Application design pattern (aka Single Page Interface)
  - Optimal startup-time, fast view switching, quick page navigation, request minimization
- Enforces XPages Mobile best practices:
  - Uses Custom Controls for each Application Page
  - Uses most mobile-friendly XPages controls like Data View, Form Table etc
  - Provides “Back” button and other app navigation options
- Supports bleeding edge features like infiniteScroll
- Designed with Future Extensibility in mind
  - More data sources may be supported in future (for enterprise and social data)
  - Options in place to support other controls for documents and document collections



## Creating your own Designer Wizard

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Creating a Designer Wizard

- This section of the presentation is aimed at advanced XPages developers that create and deploy their own XPages controls.
- Prerequisites
  - You know how to create an XPages control
    - [http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Creating\\_a\\_Java\\_Control\\_in\\_an\\_NSF](http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Creating_a_Java_Control_in_an_NSF)
  - You know how to package the control in an XPages library
    - [http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Creating\\_an\\_XPages\\_Library](http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Creating_an_XPages_Library)
  - You know how to deploy an XPages library to Designer and Domino Server
    - [http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Installing\\_an\\_XPages\\_Library\\_into\\_Designer](http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Installing_an_XPages_Library_into_Designer)
    - [http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Deploying\\_XPage\\_Libraries](http://www-10.lotus.com/ldd/ddwiki.nsf/dx/Deploying_XPage_Libraries)

# Creating a Designer Wizard

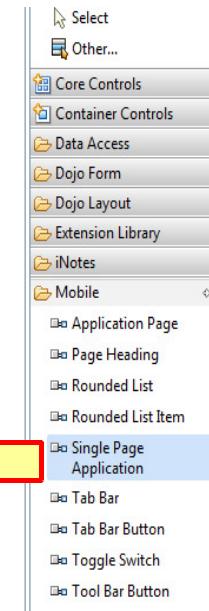
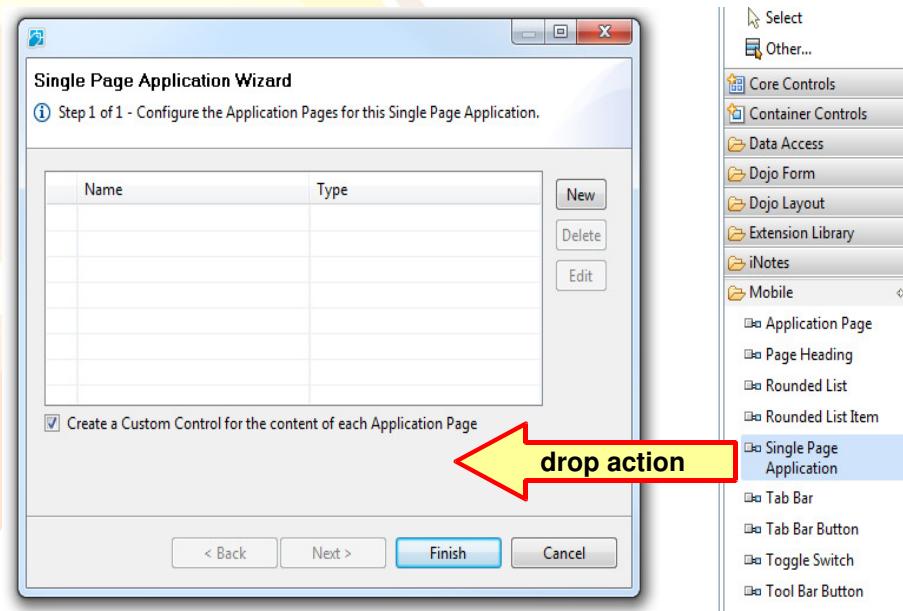
## Overview

- The target control must be contributed to Designer by an XPages library (plugin)
  - A Wizard cannot be implemented for a control that resides in an NSF
- To create a Wizard we must create a separate Designer “tooling” plugin for our control
  - If we package the Wizard in the same plugin as the control then we can't deploy the control to a Domino Server as it will have dependencies on Designer plugins that don't exist on the Server
- The tooling plugin must contribute to the **“com.ibm.designer.domino.xsp.editor.component”** extension point.
  - This allows us to specify a “drop action” for our control
- We use the Java perspective in Designer to create plugins

# Creating a Designer Wizard

## Fundamentals

- By intercepting the Drop Action, we can display our Wizard when the control is dragged onto an XPage
- We then have direct access to the XSP root element for the control
- Depending on the options selected by the user in the Wizard, we can then add/modify the attributes of the root element and add child elements if required
- This gives us complete control over the final XSP markup that is placed on the XPage for the control



# Creating a Designer Wizard

## Demo - Background

- For demo purposes I'm using the control from the Mastering XPages book, Chapter 12 XPages Extensibility. This chapter shows how to create an XPages control in an NSF
- The control is called MXPD Spinner. I've taken this control and packaged it in an XPages library (plugin), so that it can be used across multiple NSFs and deployed to my domino Server. This, of course, also allows me to create a Wizard for the control.
- The control implements a simple spinner consisting of a text box and two buttons



- In the demo I will create a Wizard for the MXPD Spinner that will allow it to be configured as it's being dropped onto the XPage

# Creating a Designer Wizard

## Demo - Create a new tooling plugin

- The first thing to do is to create a new MXPD “tooling” plugin for our control
  - `com.acme.mxpd.tooling`
- Add the following dependencies
  - `com.ibm.commons`
  - `com.ibm.xsp.core`
  - `org.eclipse.jface`
  - `com.ibm.designer.domino.xsp.editor`
  - `com.ibm.designer.domino.ide.resources`
  - `com.ibm.xsp.extlib.designer.tooling`
- Add the plugin to the “com.acme.mxpd.feature” feature so that it can be installed to Designer

# Creating a Designer Wizard

## Demo - Create the extension

- Create a Designer extension in “plugin.xml” so that we can intercept the Drop Action

```
<extension point="com.ibm.designer.domino.xsp.editor.component">
  <components
    id="Acme Controls"
    schemaUri="http://www.acme.com/xsp">
    <component tagName="uiSpinner">
      <palettesItem
        action="com.acme.mxpd.tooling.MxpdDropAction"
        label="MXPD Spinner"
        visible="true">
      </palettesItem>
    </component>
  </components>
</extension>
```

# Creating a Designer Wizard

## Demo - Create the Drop Action Class

- It must extend the *XpagesPaletteDropActionDelegate* class and override the *createElement* function

```
public class MxpdDropAction extends XpagesPaletteDropActionDelegate {  
  
    @Override  
    protected Element createElement(Document doc, String prefix) {  
        Shell shell = getControl().getShell();  
  
        // Create the base element  
        Element element = super.createElement(doc, prefix);  
  
        // Launch the Wizard Dialog  
        WizardDialog dialog = new WizardDialog(shell, new MxpdWizard(shell, element, getDesignerProject().getFacesRegistry()));  
        if (WizardDialog.OK != dialog.open()) {  
            return null;  
        }  
  
        return element;  
    }  
}
```

# Creating a Designer Wizard

## Demo - Create the Wizard class

- It must extend the standard jface Wizard class

```
public class MxpdWizard extends Wizard {  
  
    private Element rootEl;  
    private FacesRegistry facesRegistry;  
  
    public MxpdWizard(Shell shell, Element rootEl, FacesRegistry facesRegistry) {  
        super();  
        this.rootEl = rootEl;  
        this.facesRegistry = facesRegistry;  
    }  
  
    @Override  
    public boolean performFinish() {  
        return false;  
    }  
}
```

# Creating a Designer Wizard

## Demo – Flesh out the Wizard with pages

- Override the *addPages* method to add our Wizard Pages

```
private MxpdWizardPageOptions optionsPage = new MxpdWizardPageOptions(WIZARD_TITLE);  
private MxpdWizardPageSteps stepsPage = new MxpdWizardPageSteps(WIZARD_TITLE);
```

```
@Override  
public void addPages() {  
    addPage(optionsPage);  
    addPage(stepsPage);  
}
```

- Create the classes for two our Wizard Pages, they must extend the jface WizardPage class. These Wizard Pages add SWT UI controls to the Wizard in the overridden *createControl* function
  - MxpdWizardPageOptions
  - MxpdWizardPageSteps

# Creating a Designer Wizard

## Demo – Generate the markup

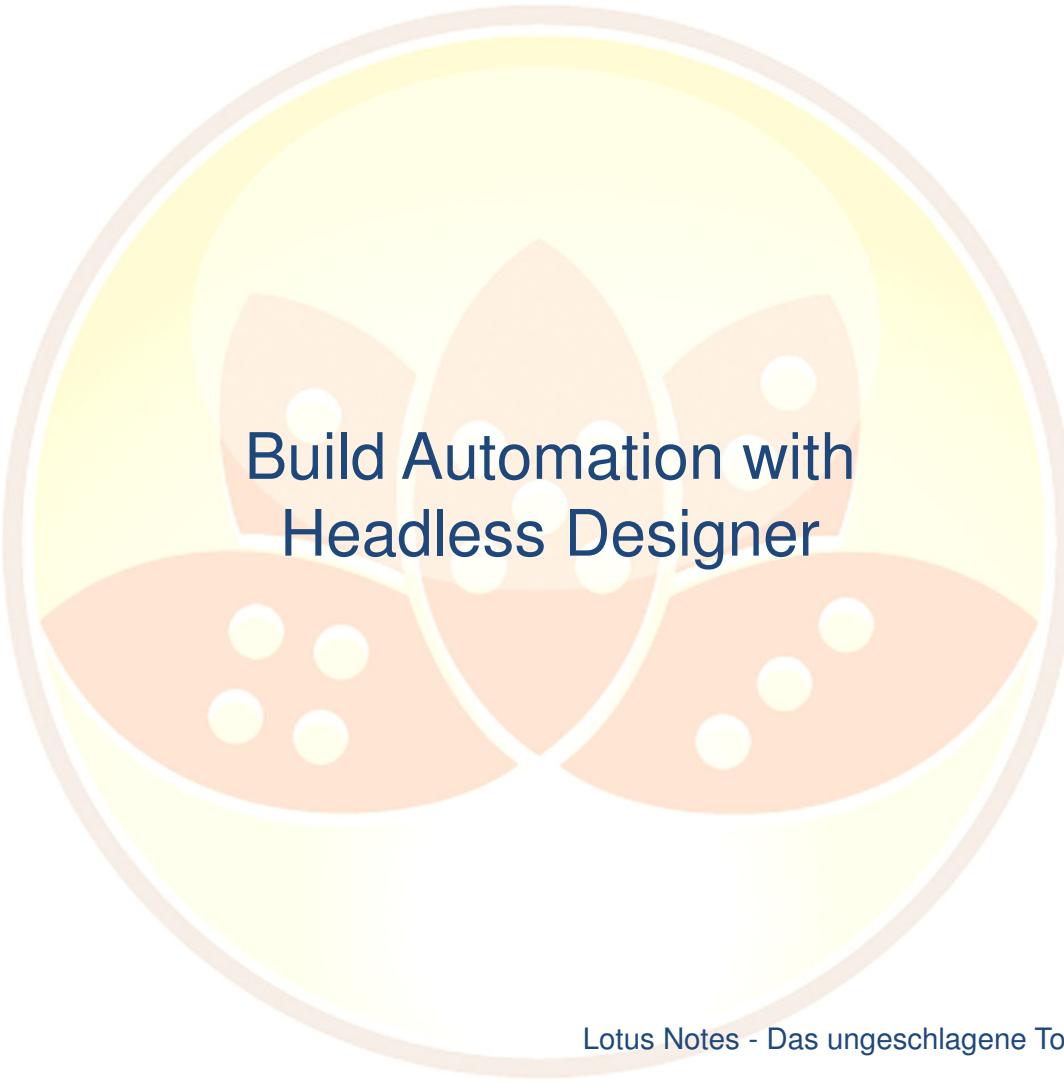
- Create a generateMarkup function and call it from the *performFinish* method. Use Designer APIs to manipulate the W3C DOM element and augment the XSP markup to be placed onto the XPage

```
private void generateMarkup() {  
    Document doc = rootElement.getOwnerDocument();  
  
    // Set main properties  
    FormModelUtil.setAttribute(rootElement, "max", optionsPage.getMax());  
    FormModelUtil.setAttribute(rootElement, "min", optionsPage.getMin());  
    FormModelUtil.setAttribute(rootElement, "value", optionsPage.getInitial());  
  
    // Complex property - Create child elements  
    Element stepSizes = FormModelUtil.createElement(doc, facesRegistry, NAMESPACE, "this.stepSizes");  
    Element largeSmallStep = FormModelUtil.createElement(doc, facesRegistry, NAMESPACE, "largeSmallStep");  
  
    // Set complex attrs  
    FormModelUtil.setAttribute(largeSmallStep, "largeStep", stepsPage.getLargeStep());  
    FormModelUtil.setAttribute(largeSmallStep, "smallStep", stepsPage.getSmallStep());  
  
    // Add child elements  
    stepSizes.appendChild(largeSmallStep);  
    rootElement.appendChild(stepSizes);  
}
```

# Creating a Designer Wizard

## Why create wizards

- Wizards simplify complex development tasks and save time
  - The Single Page Application Wizard produces a feature rich application without any coding experience required
    - It assembles and configures multiple controls
    - It creates custom controls
    - It enforces best practice design patterns
    - The source code is freely available on OpenNTF



EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Headless Designer (9.0.1 Tech Preview)

Making Designer work for you when you're not IN work!

- Problem
  - You've a group of developers working on an application
  - They're using a source code repository
  - You want to produce a daily build of the application (NSF)
  - HOW?
- Solution
  - Headless Designer to the rescue!
  - Use a Task Scheduler to execute a daily script
  - This script takes a snapshot of the source repository and then invokes Headless Designer to produce the NSF

# Headless Designer \*

## What is it?

- Headless Designer is a command line build system.
  - Its main function is to produce an NSF from an “on disk” project
- The command line build can be run in two ways
  - Single command
    - Suitable for simple use cases. ie. Building a single NSF
  - Command file
    - Enables more complex use cases involving multiple NSF builds with dependancies

(\* 9.0.1 Tech Preview)

# Headless Designer \*

notes.ini

- To allow Designer to be invoked “headlessly” there is a setting in “notes.ini”
  - DESIGNER\_AUTO\_ENABLED=true

(\* 9.0.1 Tech Preview)

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

EntwicklerCamp 2014

# Headless Designer \*

## Single command

- The single command has the format:
  - designer.exe -RPARAMS -vmargs -Dcom.ibm.designer.cmd=<single command>
- The single command is made up of a number of comma separated components, and is contained within double quotes
  - "<clean>,<exit>,<file name>,<job name>,<file path>,<file name>"
    - clean: true/false
    - exit: true/false
    - file name: The name of the NSF/NTF to be created. If no file extension is included then an NSF will be created.
    - job name: the job to be run
    - file path: the path to the .project file of the "on-disk" project to be built

(\* 9.0.1 Tech Preview)

# Headless Designer \*

Single command

- Example:

- designer.exe -RPARAMS -console -vmargs -Dcom.ibm.designer.cmd="true,true,Discussion.nsf,importandbuild,C:\XPageApps\Discussion\.project,Discussion.nsf"

```
2014/01/15 13:38:40.450 INFO Importing ondisk C:\XPageApps\DemoApp\.project and  
creating NSF Discussion.nsf ::class.method=com.ibm.designer.domino.tools.userles  
sbuild.jobs.ImportAndBuildJob.<init>() ::thread=main ::loggername=com.ibm.design  
er.headless.jobs  
2014/01/15 13:38:40.458 INFO creating NSF Discussion.nsf [server:] ::class.method  
d=com.ibm.designer.domino.tools.userlessbuild.jobs.CreateNSFJob.<init>() ::threa  
d=Worker-11 ::loggername=com.ibm.designer.headless.jobs  
2014/01/15 13:38:44.296 INFO Starting Job - AssociateOnDiskWithNSF ::class.method  
d=com.ibm.designer.domino.tools.userlessbuild.jobs.ImportOnDiskProjectJob.run()  
::thread=Worker-11 ::loggername=com.ibm.designer.headless.jobs  
2014/01/15 13:38:50.435 INFO -- executing 1421 syncOps ::class.method=com.ibm.de  
signer.domino.tools.userlessbuild.jobs.AssociateOnDiskWithNSFJob$NoUISyncAction$  
1.run() ::thread=Worker-11 ::loggername=com.ibm.designer.headless.jobs
```

Import

Create

Associate

Sync

(\* 9.0.1 Tech Preview)

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Headless Designer \*

## Command file

- For multiple commands, it is possible to pass a file path, as a Java property, to the command line build.
  - designer -RPARAMS -vmargs -Dcom.ibm.designer.cmd.file=<path to command file>
- This file contains one command per line, example:

```
config,true,true
importandbuild,C:\test\XPagesExt\.project,testXPages
wait,testXPages,20
importandbuild,C:\test\XPagesSBT\.project,testSBT
wait,testSBT,20
clean
exit
```

(\* 9.0.1 Tech Preview)

# Headless Designer \*

## Command file jobs

- importandbuild <on disk project path> <nsf name>
  - imports the “on-disk” project, creates the NSF, and begins the association. Processing of the next job does not wait for the sync or build to finish
- wait <project name> <# of retries>
  - Waits for the given project to complete all build or sync operations
- config <stop on error - true/false> <exit on error - true/false>
  - allows the configuring of some properties that control operation
- clean
  - deletes all projects created so far in this execution. Does NOT delete the project contents
- exit [# milli delay] [restart-true/false] [resume - true/false]
  - Exits Designer. restart will restart Designer, and resume will continue processing the command file if there are commands after this exit command

(\* 9.0.1 Tech Preview)

# Headless Designer \*

## Update site job \*\*

- Advanced job to be used when the NSF has a dependency on one or more update sites
  - Eliminates the need to manually install update sites
- `updatemanagerjob -command <cmd> [-from jar:file:<updatesite>] -to file:<dest> -featureId <feature> -version <ver>`
  - cmd: install / enable / disable / uninstall
  - updatesite: path to updatesite zip file (only for install)
  - dest : destination path, normally notes workspace
  - feature: feature to act on
  - ver: version of feature to act on
- Example:
  - `updatemanagerjob,-command install -from jar:file:C:/updatesite.zip!/ -to file:C:/Notes/Data/workspace/applications -featureId com.ibm.sbt.domino.playground.feature -version 1.0.0.20131121-0623`

(\* 9.0.1 Tech Preview)

(\*\* This feature is NOT in version 9.0.1 but is intended for future release)

# Headless Designer \*

## Scripting and Scheduling

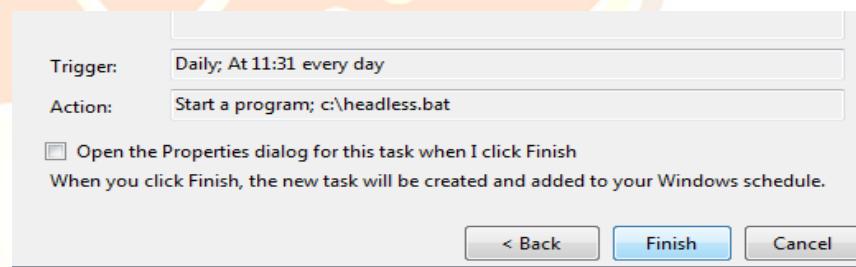
- Sample Script File: headless.bat

```
cd Notes  
start /wait designer.exe -RPARAMS -console -vmargs -D%command%=%build%

- %command%: com.ibm.designer.cmd.filecom.ibm.designer.cmd.file
- %build%: path to command file

```

- Scheduling
  - Use a Task Scheduler to invoke the script on a periodic basis



(\* 9.0.1 Tech Preview)



## Q & A

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren



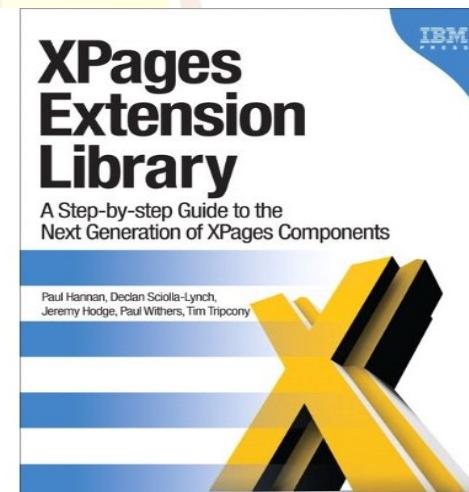
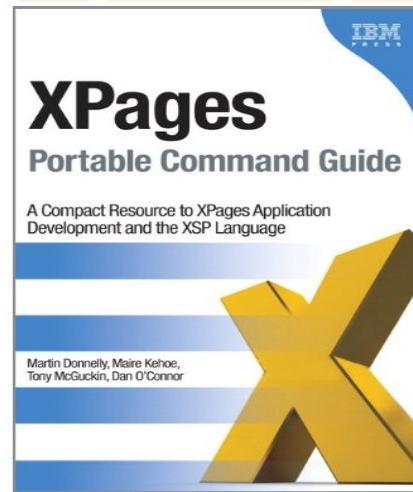
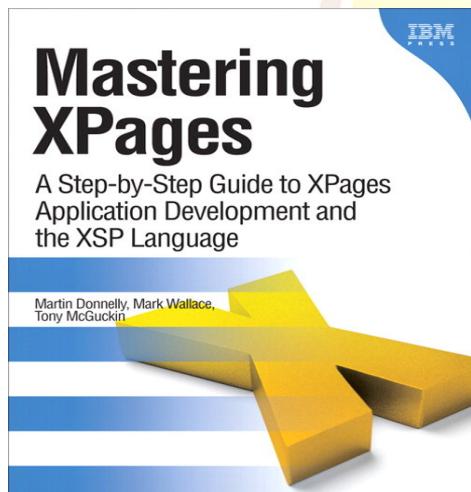
Resources

EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Technical Education

- IBM Press Books and eBooks
  - Three major publications over the past two years
  - All available for evaluation in the bookstore in the Solutions Expo



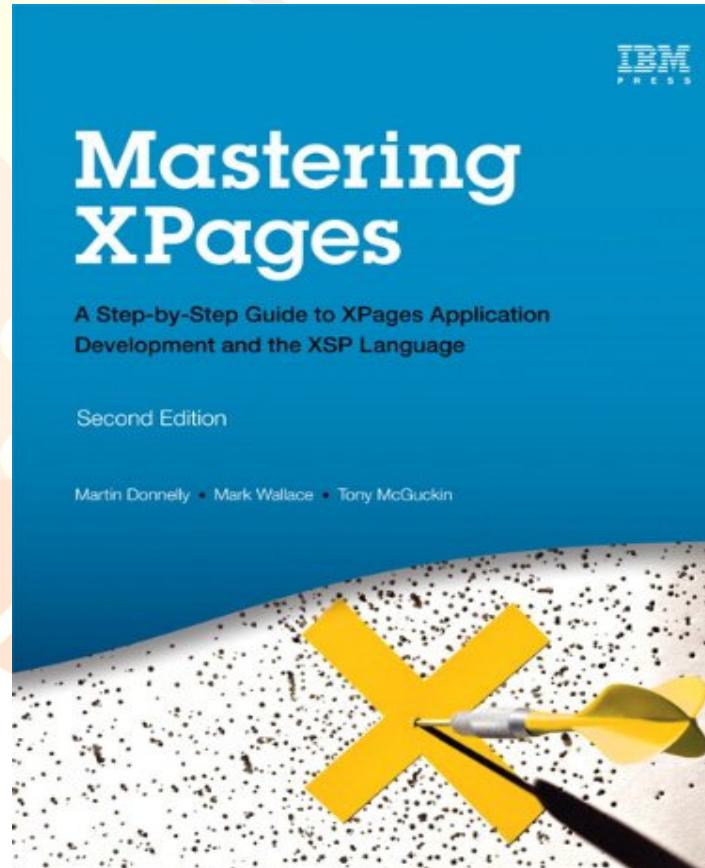
EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

# Technical Education

- Coming very soon ...
- Mastering XPages 2<sup>nd</sup> Edition

- Based on Notes/Domino 9.0.1
- 4 new chapters
  - Advanced Performance/Scalability
  - Mobile Application Development
  - Deep Dive on Debugging
  - All About Application Layout
- Comprehensive updates to all pre-existing 1<sup>st</sup> edition content
- Hardcover
  - ~500 pages of new content
  - ~1200 pages total



EntwicklerCamp 2014

Lotus Notes - Das ungeschlagene Tool der Zukunft seit fast 20 Jahren

## More Information – General

- For all information on XPages head to Xpages.info  
<http://xpages.info/XPagesHome.nsf/Home.xsp>
- The Notes Domino Application Development Wiki  
<http://www-10.lotus.com/ldd/ddwiki.nsf>
- Free XPages snippets from the community  
<http://openntf.org/XSnippets.nsf/home.xsp>
- And don't forget the XPages books from IBM Press  
<http://www.ibmpressbooks.com/search/index.asp?query=xpages>
- Learning Roadmaps
  - [\*\*New Developers\*\*](#)
  - [\*\*Experienced Developers\*\*](#)

# More Information – Summary

- XPages.info – One Stop Shopping for XPages  
<http://xpages.info>
- XPages Forum – Got Questions, Need Answers?  
<http://xpages.info/forum>
- OpenNTF – Open Source Community  
<http://www.openntf.org>
- Domino Application Development Wiki  
<http://www.lotus.com/ldd/ddwiki.nsf>
- XPages Blog  
<http://xpagesblog.com>
- IBM Educational Offerings  
<http://www.ibm.com/software/lotus/training/n8deducationofferings.html>



# More Information – Training

- TLCC Classes - <http://www.tlcc.com/admin/tlccsite.nsf/DeveloperView.xsp>
  - ND85XPGS - Developing XPages using Domino Designer 8.5
  - ND85XPG2 - XPages Development 2 for Notes and Domino 8.5
  - ND85XJSC - JavaScript for XPages Development (8.5)
  - XP85MOBL - Mobile XPages Development for Domino 8.5
- IBM Classes - [http://www-304.ibm.com/jct03001c/services/learning/ites.wss/zz/en?pageType=tp\\_search](http://www-304.ibm.com/jct03001c/services/learning/ites.wss/zz/en?pageType=tp_search)
  - D8L51 - Fundamentals of IBM Lotus Domino 8.5.1 Application Development
  - D8L53 - Developing IBM Lotus Domino 8.5.2 Applications: Intermediate XPage Design
  - D8L54 - Developing IBM Lotus Domino 8.5.2 Applications: Advanced XPage Design
  - D8L55 - Introduction to IBM Lotus Domino 8.5 XPages
  - D8L56 - Modernizing IBM Lotus Domino 8.5.2 Applications
  - D8P51 - Fundamentals of IBM Lotus Domino 8.5.1 Application Development (SPVC)
  - D8P53 - Developing IBM Lotus Domino 8.5.2 Applications: Intermediate XPage Design (SPVC)
  - D8P54 - Developing IBM Lotus Domino 8.5.2 Applications: Advanced XPage Design (SPVC)
  - D8P55 - Introduction to IBM Lotus Domino 8.5 Xpages (SPVC)
  - D8P56 - Modernizing IBM Lotus Domino 8.5.2 Applications (SPVC)
  - D8S44 - Developing XPages using Domino Designer 8.5
  - D8S61 - XPage Development 2 for Notes and Domino 8.5
  - D8S60 - XPages for Notes and Domino 8.5 Package
  - D8S46 - XPages and JavaScript for Domino 8.5 Package

# More Information

- IBM Certification
  - IBM Certified Application Developer - Lotus Notes and Domino 8.5  
<http://www-03.ibm.com/certify/certs/14003105.shtml>
  - IBM Certified Advanced Application Developer - Lotus Notes and Domino 8.5  
<http://www-03.ibm.com/certify/certs/14003205.shtml>
  - Test LOT-922: Developing IBM Lotus Domino 8.5.2 Applications: Advanced XPage Design  
<http://www-03.ibm.com/certify/tests/ovrL922.shtml>
- Mobile Design Info
  - Mobile Web Application Best Practices  
<http://www.w3.org/TR/mwabp/>
  - SPA/SPI Pattern  
[http://en.wikipedia.org/wiki/Single-page\\_application](http://en.wikipedia.org/wiki/Single-page_application)
  - The Single Page Interface Manifesto  
[http://itsnat.sourceforge.net/php/spim/spi\\_manifesto\\_en.php](http://itsnat.sourceforge.net/php/spim/spi_manifesto_en.php)



**Mobile Web Application Best Practices**

# Engage Online

- **SocialBiz User Group** [socialbizug.org](http://socialbizug.org)
  - Join the epicenter of Notes and Collaboration user groups
- **Follow us on Twitter**
  - [@IBMConnect](https://twitter.com/IBMConnect) and [@IBMSocialBiz](https://twitter.com/IBMSocialBiz)
- **LinkedIn** <http://bit.ly/SBComm>
  - Participate in the IBM Social Business group on LinkedIn:
- **Facebook** <https://www.facebook.com/IBMSocialBiz>
  - Like IBM Social Business on Facebook
- **Social Business Insights blog** [ibm.com/blogs/socialbusiness](http://ibm.com/blogs/socialbusiness)
  - Read and engage with our bloggers

# Acknowledgements and Disclaimers

**Availability.** 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.

The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation 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.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. 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.

**© Copyright IBM Corporation 2014. All rights reserved.**

▪ **U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.**

▪ IBM, the IBM logo, ibm.com, Domino and Notes are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)

SubversionEdge is a trademark or registered trademark of CollabNet.

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