Connect your Lotus Notes app to the Activity Stream with XPages

Frank van der Linden

e-office

Agenda

- Introduction
- Social Business
- oAuth and OpenSocial
- Let’s connect to the Activity Stream
- Post to the Activity Stream
- Delete from the Activity Stream
- Q&A
- Usefull links

Who is Frank van der Linden

- I live in Utrecht in the Netherlands.
- My role is XPages/Domino/Web developer at e-office since 2000. So I started with Lotus Notes 4.5.x.
- I develop XPages application since the introduction of XPages in Lotus Notes 8.5.0.
- In my spare time I do a lot of running and then I mean a lot.
- And I am married and have 2 daughters

e-office

And he works at e-office

- Celebrated in 2011 it’s 20ste anniversary
- First Lotus Business Partner in the Netherlands
- E-office is IBM Premier Business Partner, Microsoft Gold Partner and RIM Alliance Elite partner

e-office

THEHUMANSOFTWAREORGANISATION

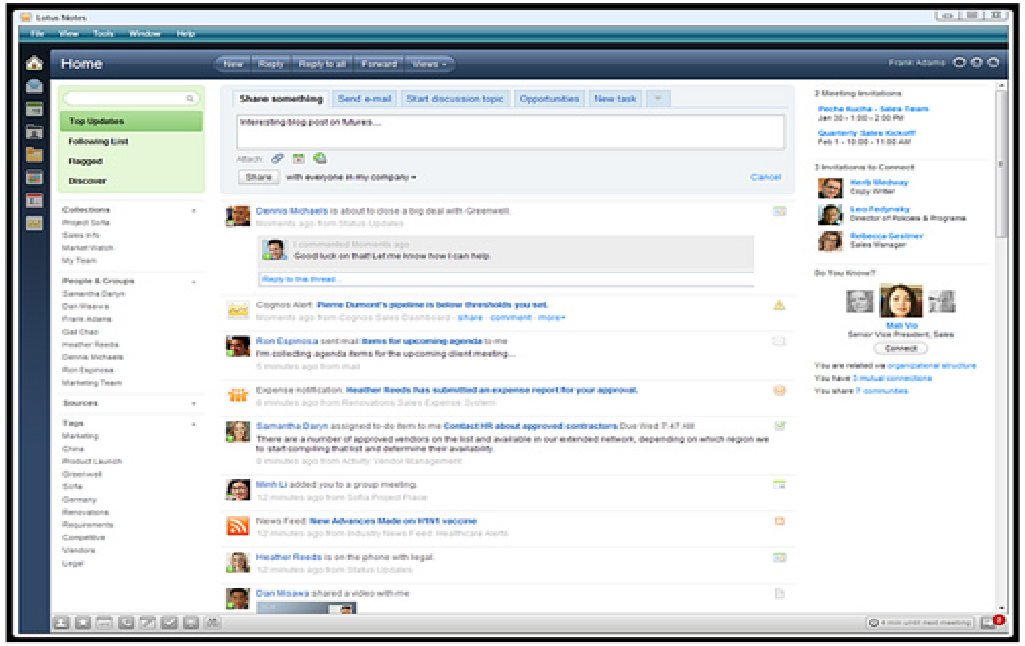
Agenda

- Introduction
- Social Business Toolkit
- oAuth and OpenSocial
- Let’s connect to the Activity Stream
- Post to the Activity Stream
- Delete from the Activity Stream
- Q&A
- Usefull links

e-office

Social Business Toolkit

- Manage all your daily business in one stream
- Also your mail and to-do's
- Easy to connect by providing API's
- It will be integrated in IBM Connections Next and Lotus Notes Next



Interesting blog post on futures...

Dennis Michaels is about to close a big deal with Greenwell.
Moments ago from Status Updates
I commented Moments ago
Good luck on that! Let me know how I can help.
Reply to this thread...

Cognos Alert: Pierre Dumont's pipeline is below thresholds you set.
Moments ago from Cognos Sales Dashboard - share - comment - more

Ron Espinosa sent mail Items for upcoming agenda to me
I'm collecting agenda items for the upcoming client meeting...
5 minutes ago from mail

Expense notification: Heather Reeds has submitted an expense report for your approval.
8 minutes ago from Renovations Sales Expense System

Samantha Daryn assigned to-do item to me Contact HR about approved contractors Due Wed 7:47 AM
There are a number of approved vendors on the list and available in our extended network, depending on which region we to start compiling that list and determine their availability.
8 minutes ago from Activity: Vendor Management

Minh Li added you to a group meeting.
12 minutes ago from Sofia Project Place

News Feed: New Advances Made on H1N1 vaccine
12 minutes ago from Industry News Feed: Healthcare Alerts

Heather Reeds is on the phone with legal.
12 minutes ago from Status Updates

Dan Misawa shared a video with me

2 Meeting Invitations
Pecha Kucha - Sales Team
Jan 30 - 1:00 - 2:00 PM
Quarterly Sales Kickoff
Feb 1 - 10:00 - 11:00 AM

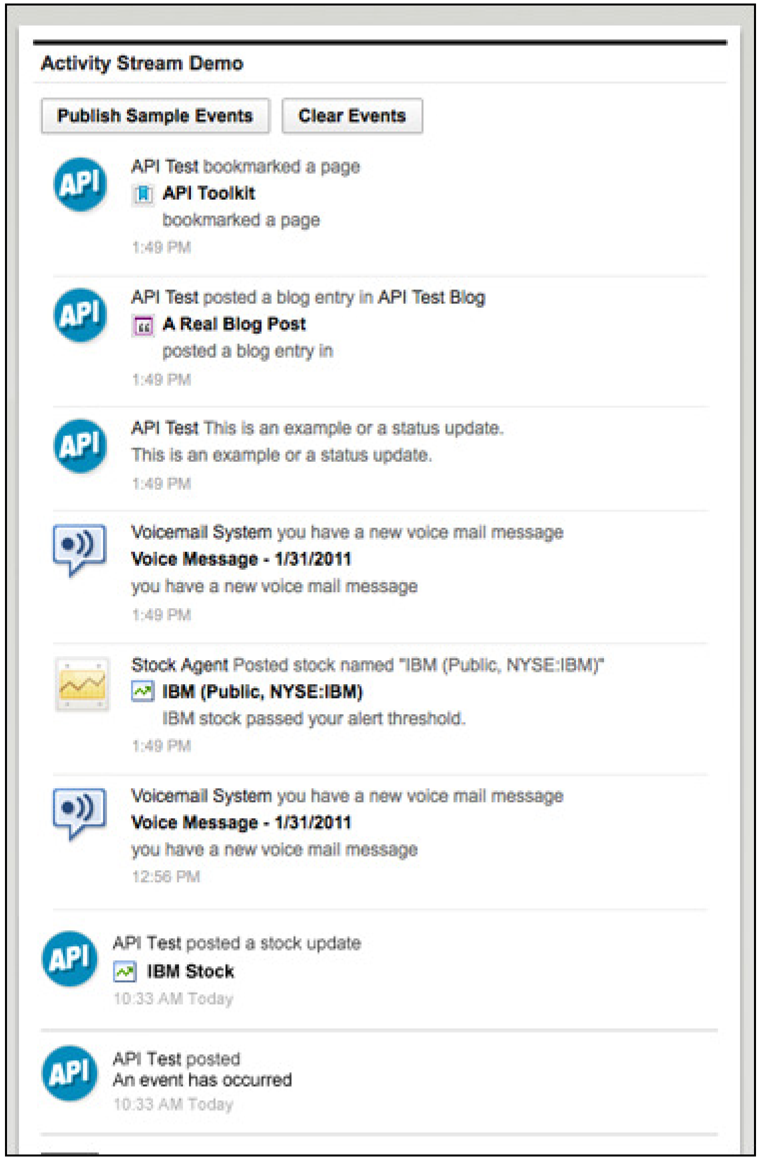
3 Invitations to Connect
Herb Medway
Copy Writer
Leo Fedynsky
Director of Policies & Programs
Rebecca Gestner
Sales Manager

Do You Know?
Mali Vo
Senior Vice President, Sales
Connect
You are related via organizational structure
You have 3 mutual connections
You share 7 communities

e-office

Activity Stream - intro

- It is part of the Social business Toolkit
- It is the stream of all information
- API's to connect to the stream
- It will part of IBM Connections Next and Lotus Notes Social Edition.
- oAuth authentication.
- Support of OpenSocial gadget specification.


Activity Stream Demo

API Test bookmarked a page
API Toolkit
bookmarked a page
1:49 PM

API Test posted a blog entry in API Test Blog
A Real Blog Post
posted a blog entry in
1:49 PM

API Test This is an example or a status update.
This is an example or a status update.
1:49 PM

Voicemail System you have a new voice mail message
Voice Message - 1/31/2011
you have a new voice mail message
1:49 PM

Stock Agent Posted stock named "IBM (Public, NYSE:IBM)"
IBM (Public, NYSE:IBM)
IBM stock passed your alert threshold.
1:49 PM

Voicemail System you have a new voice mail message
Voice Message - 1/31/2011
you have a new voice mail message
12:56 PM

API Test posted a stock update
IBM Stock
10:33 AM Today

API Test posted
An event has occurred
10:33 AM Today

e-office

Agenda

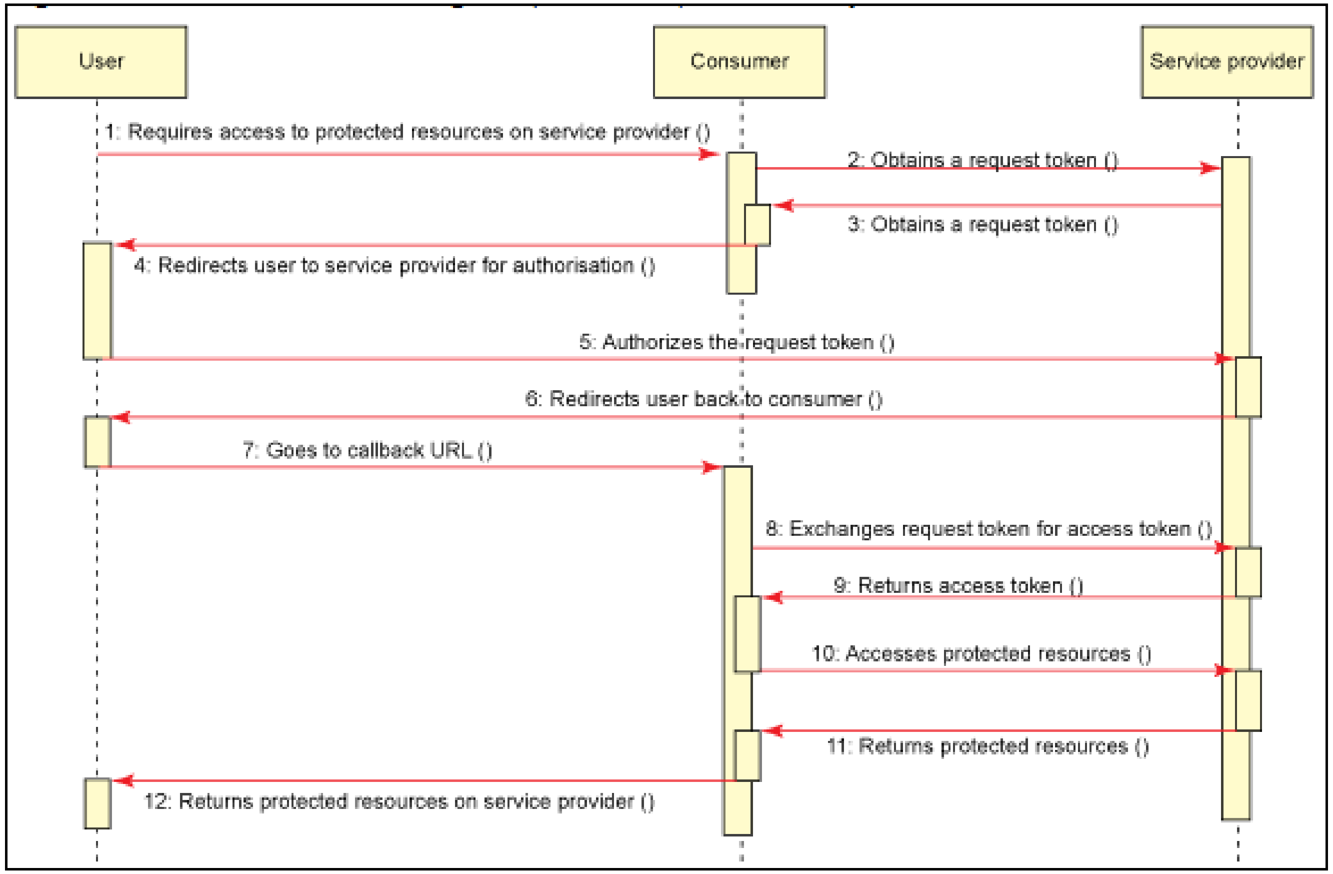
- Introduction
- Social Business Tookit
- oAuth and OpenSocial
- Let’s connect to the Activity Stream
- Post to the Activity Stream
- Delete from the Activity Stream
- Q&A
- Usefull links

What is oAuth

- OAuth (Open Authorization) is an open standard for authorization
- It allows users to share their resources stored on one site with another site without having to hand out their credentials
- OAuth allows users to hand out tokens instead of credentials to their data hosted by a given service provider.

Source: http://en.wikipedia.org/wiki/OAuth

oAuth, the 3 leg dance


User
Consumer
Service provider
1: Requires access to protected resources on service provider ()
2: Obtains a request token ()
3: Obtains a request token ()
4: Redirects user to service provider for authorisation ()
5: Authorizes the request token ()
6: Redirects user back to consumer ()
7: Goes to callback URL ()
8: Exchanges request token for access token ()
9: Returns access token ()
10: Accesses protected resources ()
11: Returns protected resources ()
12: Returns protected resources on service provider ()

e-office

oAuth in the real world

vimeo

facebook

LinkedIn

twitter

yammer

OAUTH

Dropbox

TripIt

Google Docs

OpenSocial

- Public specification that defines a component hosting environment (container)
- Based on HTML and Javascript, as well as the Google gadgets framework
- OpenSocial adopted support for Activity Streams format
- OpenSocial API and oAuth support

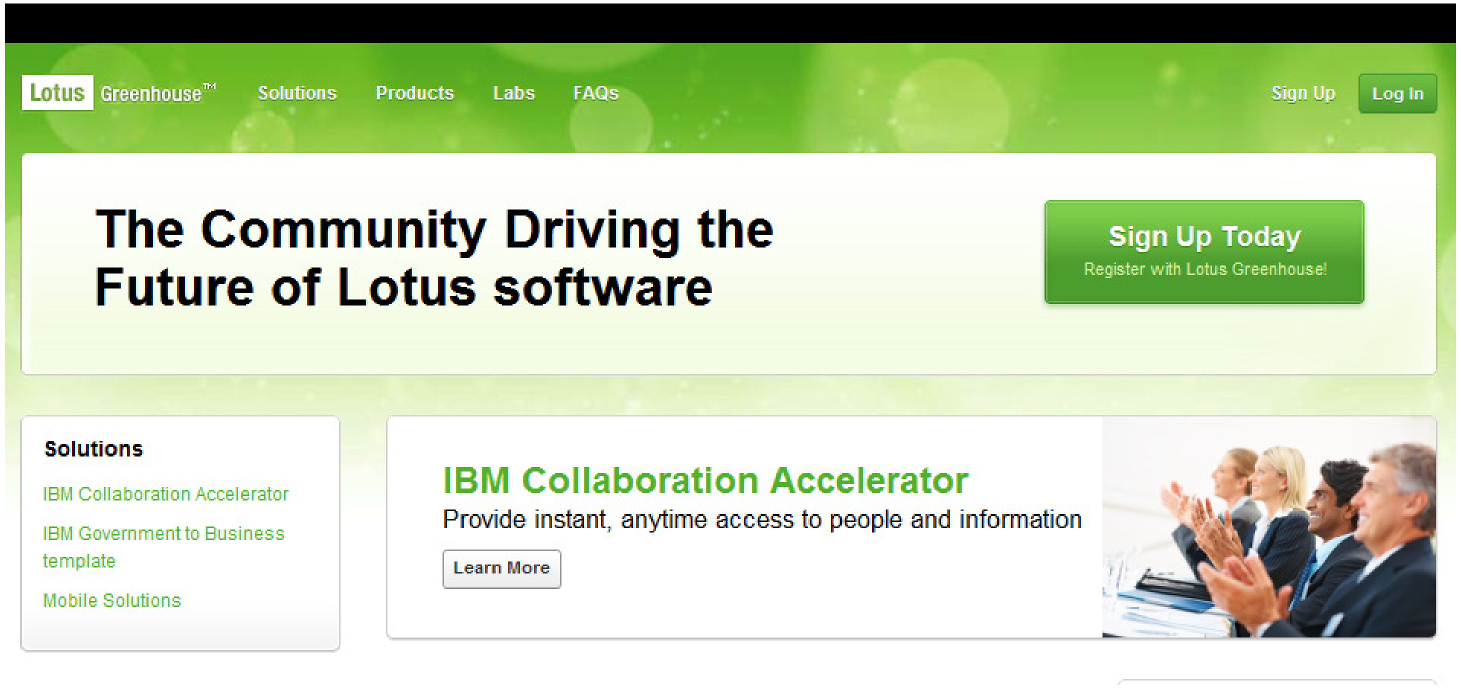
Source:http://en.wikipedia.org/wiki/OpenSocial

Agenda

- Introduction
- Social Business Toolkit
- oAuth and OpenSocial
- Let’s connect to the Activity Stream
- Post to the Activity Stream
- Delete from the Activity Stream
- Q&A
- Usefull links

Get started – get access to Greenhouse

- If you don't have an account for Greenhouse, go get it (https://greenhouse.lotus.com)



The Community Driving the Future of Lotus software

Solutions

IBM Collaboration Accelerator

IBM Government to Business template

Mobile Solutions

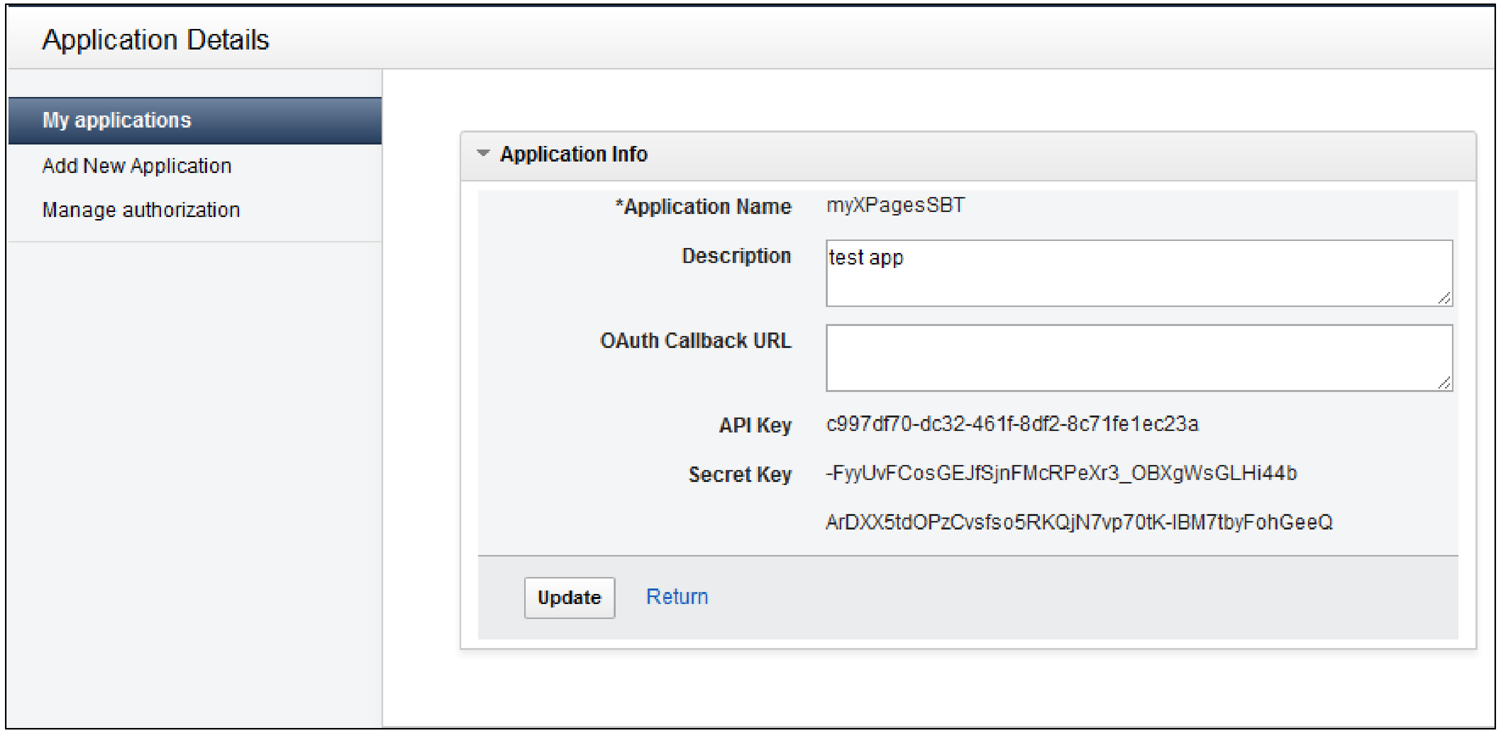
IBM Collaboration Accelerator

Provide instant, anytime access to people and information

e-office

Get started – Register your app

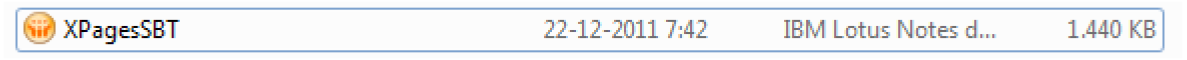
- oAuth is used, so you need to register your app.(https://greenhouse.lotus.com/vulcan/security/provider/appList?serviceProvider=vulcanToolkit)


Application Details

*Application Name	myXPagesSBT
Description	test app
OAuth Callback URL	
API Key	c997df70-dc32-461f-8df2-8c71fe1ec23a
Secret Key	-FyyUvFCosGEJfSjnFMcRPeXr3_OBXgWsGLHi44b
	ArDXX5tdOPzCvsfso5RKQjN7vp70tK-IBM7tbyFohGeeQ

Get started – Get the databases

- Get the Extension Library of OpenNTF, the 8.5.3 code stream
- Install the Extension Library on Designer and Domino server
- Deploy the Social Enabler database to your Domino server


XPagesSBT 22-12-2011 7:42 IBM Lotus Notes d... 1.440 KB

- Deploy the WebsecurityStore database to the root of your Domino server

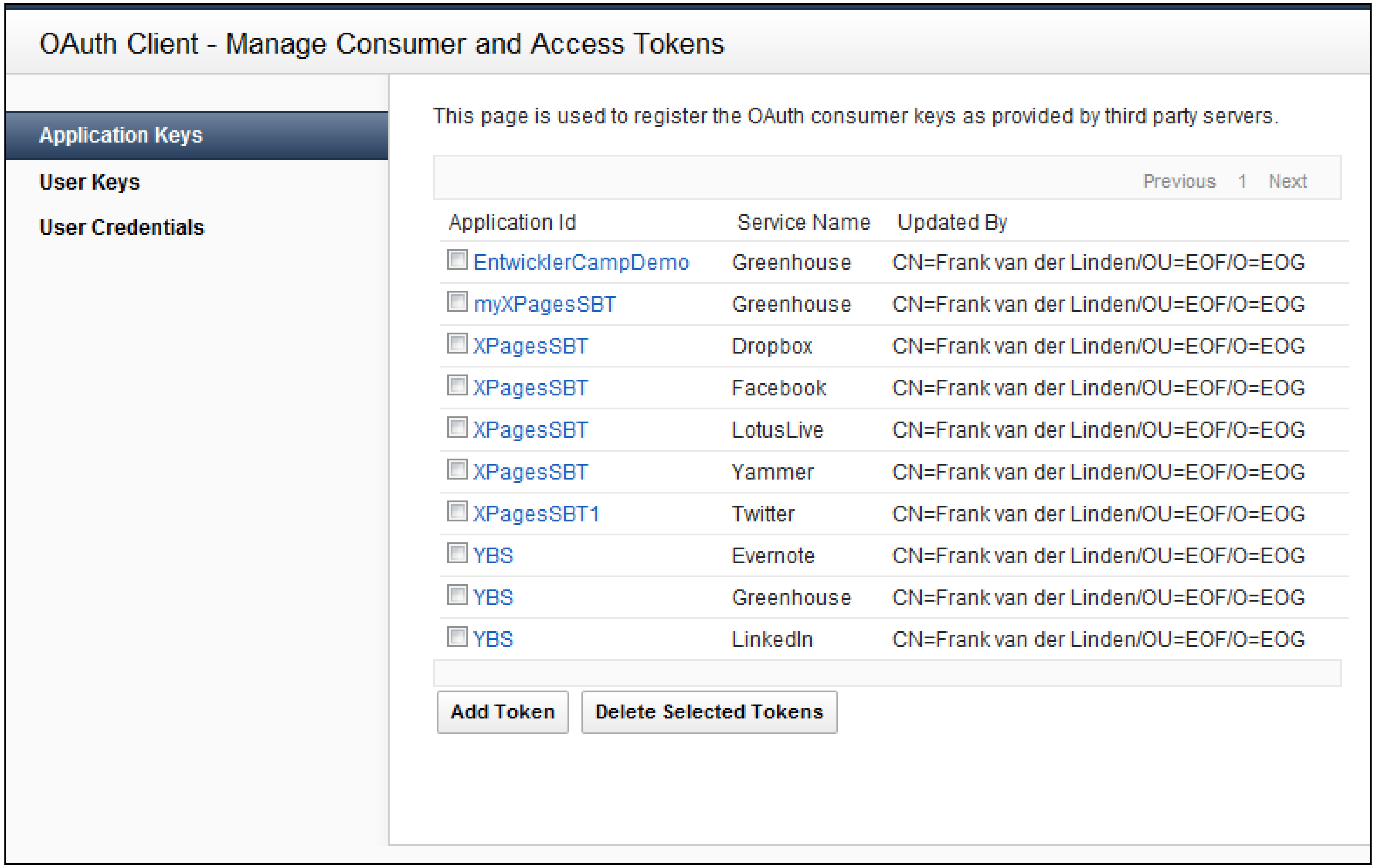
WebSecurityStore 22-12-2011 7:42 IBM Lotus Notes d... 756 KB

- And sign both databases with the correct ID.

e-office

Get started – Go to the Websecurity store

- The startpoint is KeysApplications.xsp



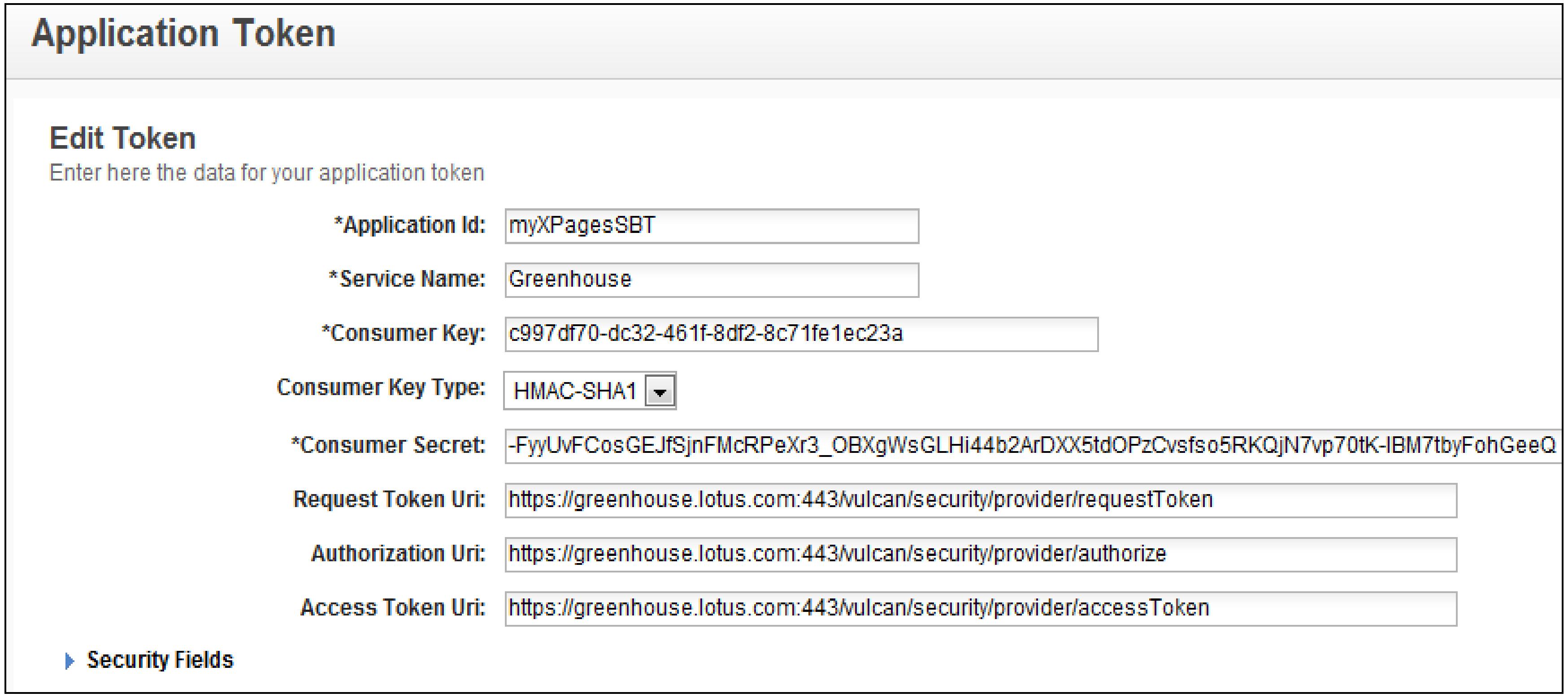
OAuth Client - Manage Consumer and Access Tokens

This page is used to register the OAuth consumer keys as provided by third party servers.

Application Id	Service Name	Updated By
☐ EntwicklerCampDemo	Greenhouse	CN=Frank van der Linden/OU=EOF/O=EOG
☐ myXPagesSBT	Greenhouse	CN=Frank van der Linden/OU=EOF/O=EOG
☐ XPagesSBT	Dropbox	CN=Frank van der Linden/OU=EOF/O=EOG
☐ XPagesSBT	Facebook	CN=Frank van der Linden/OU=EOF/O=EOG
☐ XPagesSBT	LotusLive	CN=Frank van der Linden/OU=EOF/O=EOG
☐ XPagesSBT	Yammer	CN=Frank van der Linden/OU=EOF/O=EOG
☐ XPagesSBT1	Twitter	CN=Frank van der Linden/OU=EOF/O=EOG
☐ YBS	Evernote	CN=Frank van der Linden/OU=EOF/O=EOG
☐ YBS	Greenhouse	CN=Frank van der Linden/OU=EOF/O=EOG
☐ YBS	LinkedIn	CN=Frank van der Linden/OU=EOF/O=EOG

e-office

Get started – Fill in the oAuth keys



Application Token

Edit Token

Enter here the data for your application token

*Application Id:	myXPagesSBT
*Service Name:	Greenhouse
*Consumer Key:	c997df70-dc32-461f-8df2-8c71fe1ec23a
Consumer Key Type:	HMAC-SHA1
*Consumer Secret:	-FyyUvFCosGEJfSjnFMcRPeXr3_OBXgWsGLHi44b2ArDXX5tdOPzCvsfso5RKQjN7vp70tK-IBM7tbyFohGeeQ
Request Token Uri:	https://greenhouse.lotus.com:443/vulcan/security/provider/requestToken
Authorization Uri:	https://greenhouse.lotus.com:443/vulcan/security/provider/authorize
Access Token Uri:	https://greenhouse.lotus.com:443/vulcan/security/provider/accessToken

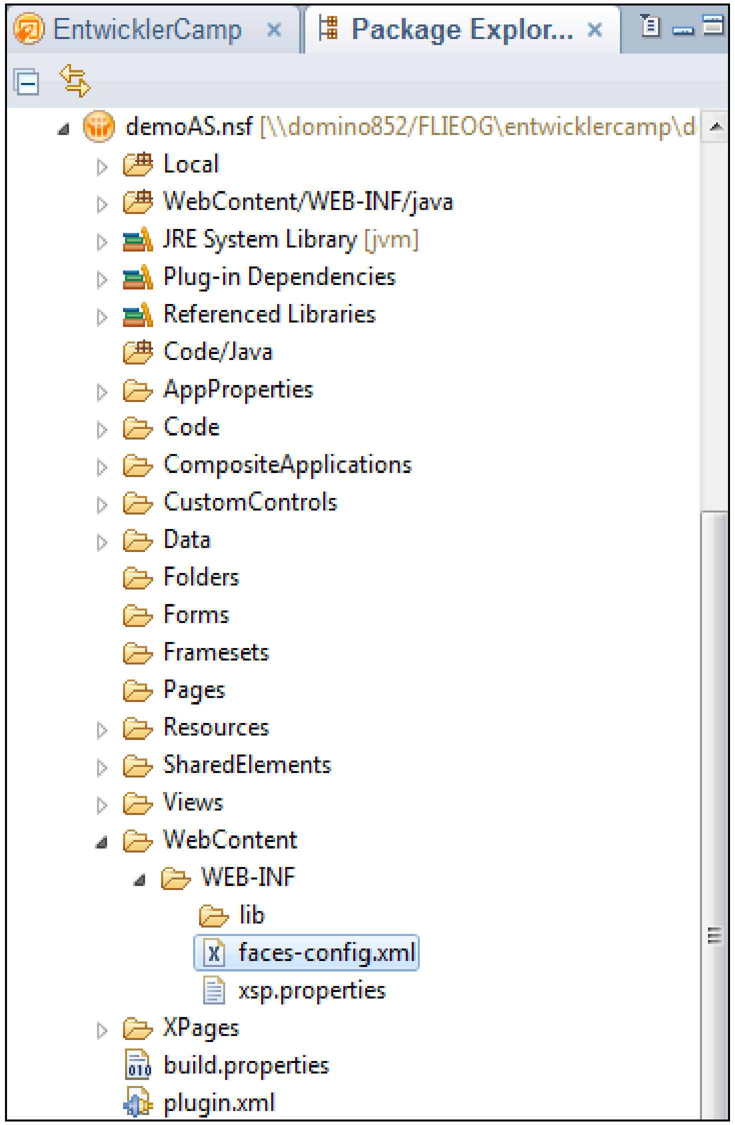
Security Fields

- https://greenhouse.lotus.com:443/vulcan/security/provider/requestToken
  https://greenhouse.lotus.com:443/vulcan/security/provider/authorize
  https://greenhouse.lotus.com:443/vulcan/security/provider/accessToken

e-office

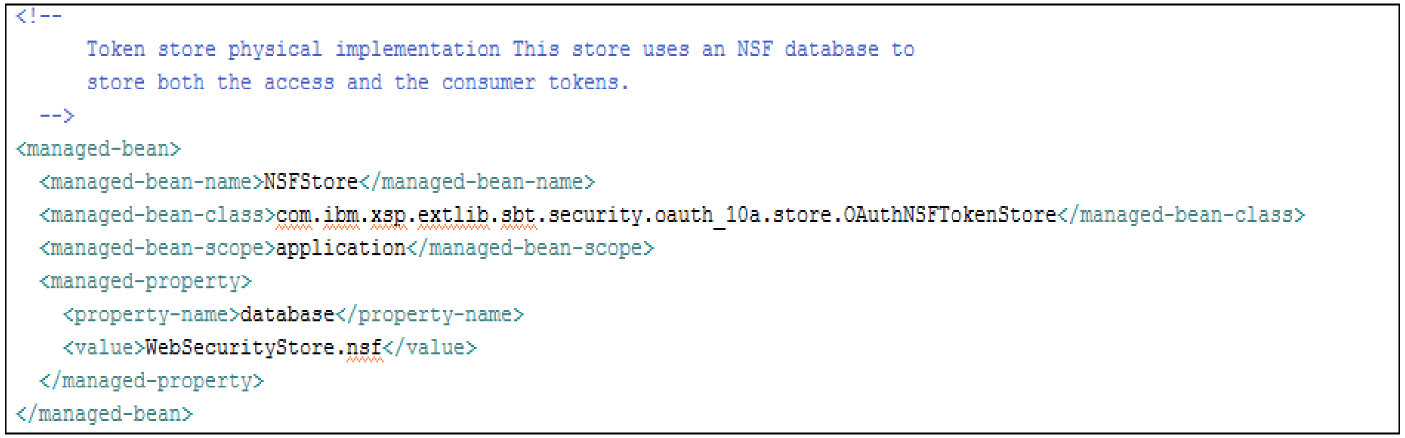
In to the code – Faces-config.xml

- This file lists bean resources and navigation rules
- It is located in the WEB-INF folder in the package explorer.



Faces-config.xml - NSFStore

- NSFStore managed bean is used for the location of the websecurity database
- It will be used by other Managed beans to store oAuth data.

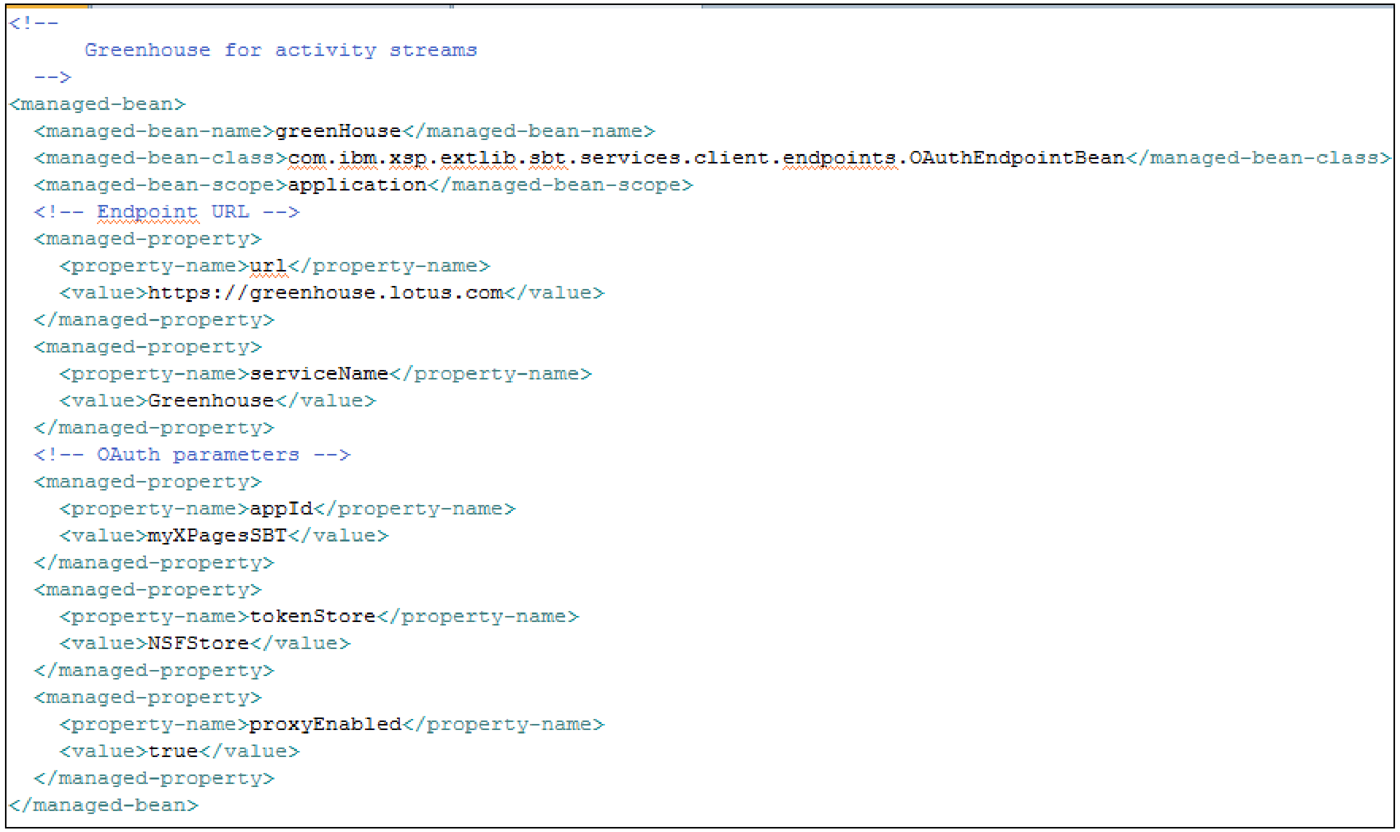


```
<!--
      Token store physical implementation This store uses an NSF database to
      store both the access and the consumer tokens.
  -->
<managed-bean>
  <managed-bean-name>NSFStore</managed-bean-name>
  <managed-bean-class>com.ibm.xsp.extlib.sbt.security.oauth_10a.store.OAuthNSFTokenStore</managed-bean-class>
  <managed-bean-scope>application</managed-bean-scope>
  <managed-property>
    <property-name>database</property-name>
    <value>WebSecurityStore.nsf</value>
  </managed-property>
</managed-bean>
```

e-office

Faces-config.xml – greenHouse managed bean

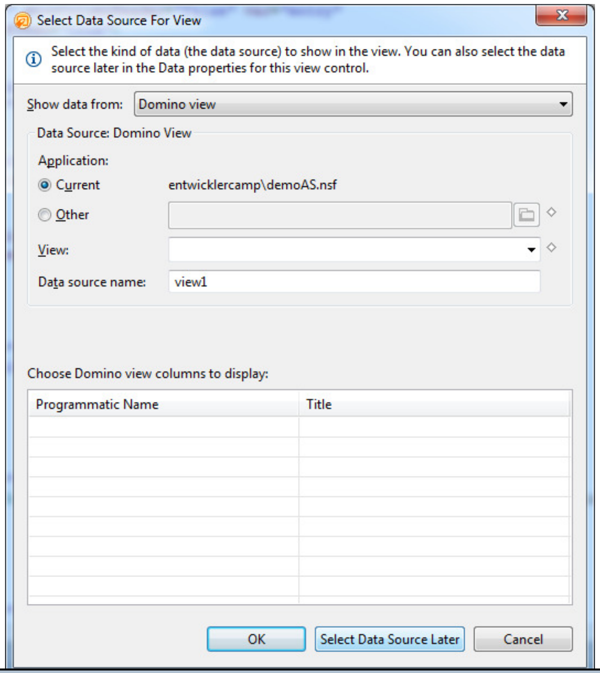
- greenHouse managed bean is used to specify the oAuthEndpoint, and were to store.
- AppId is the id, who is used as Application name in the SBT application registration page



```xml
<!--
      Greenhouse for activity streams
  -->
<managed-bean>
  <managed-bean-name>greenHouse</managed-bean-name>
  <managed-bean-class>com.ibm.xsp.extlib.sbt.services.client.endpoints.OAuthEndpointBean</managed-bean-class>
  <managed-bean-scope>application</managed-bean-scope>
  <!-- Endpoint URL -->
  <managed-property>
    <property-name>url</property-name>
    <value>https://greenhouse.lotus.com</value>
  </managed-property>
  <managed-property>
    <property-name>serviceName</property-name>
    <value>Greenhouse</value>
  </managed-property>
  <!-- OAuth parameters -->
  <managed-property>
    <property-name>appId</property-name>
    <value>myXPagesSBT</value>
  </managed-property>
  <managed-property>
    <property-name>tokenStore</property-name>
    <value>NSFStore</value>
  </managed-property>
  <managed-property>
    <property-name>proxyEnabled</property-name>
    <value>true</value>
  </managed-property>
</managed-bean>
```

How to get the entries of the Activity Stream

- Create a XPage, and name it ‘ActivityStream’
- Drop a viewpanel on this XPage
- Select Datasource later
- Goto the all properties and select as datasource the ActivityStreamData.


Select Data Source For View

Select the kind of data (the data source) to show in the view. You can also select the data source later in the Data properties for this view control.

Show data from: Domino view

Data Source: Domino View

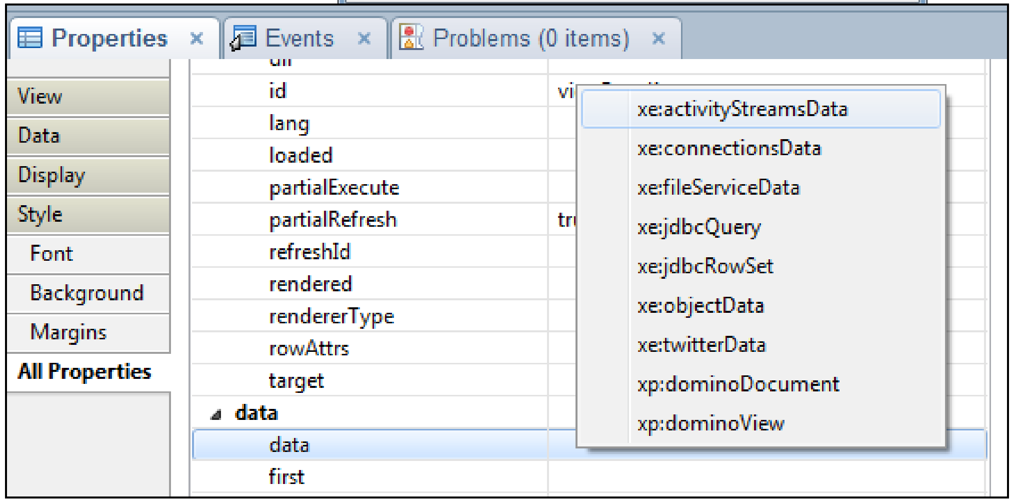
Application:
- [✓ Current] entwicklercamp\demoAS.nsf
- Other

View:

Data source name: view1

Choose Domino view columns to display:

Programmatic Name	Title

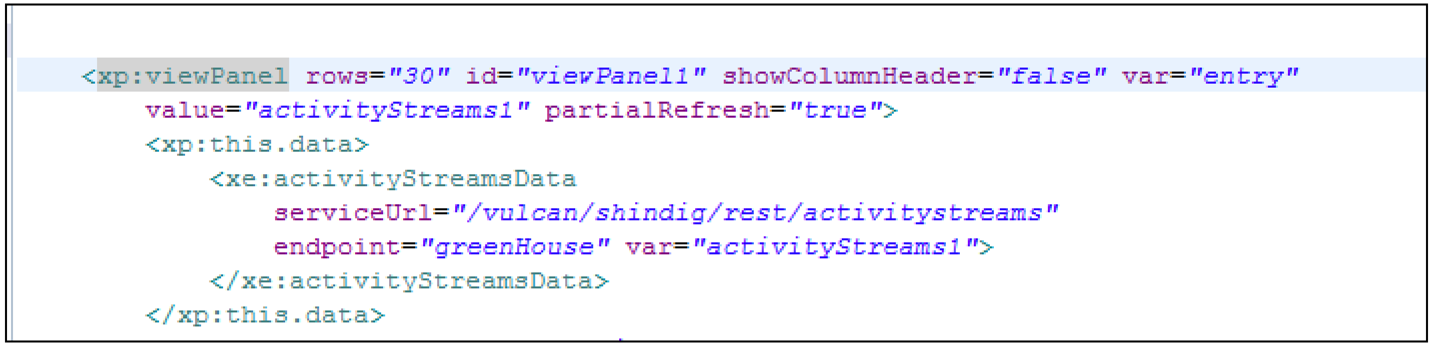


```
id
lang
loaded
partialExecute
partialRefresh
refreshId
rendered
rendererType
rowAttrs
target
data
  data
  first
```

```
xe:activityStreamsData
xe:connectionsData
xe:fileServiceData
xe:jdbcQuery
xe:jdbcRowSet
xe:objectData
xe:twitterData
xp:dominoDocument
xp:dominoView
```

How to get the entries of the Activity Stream

- The service URL: vulcan/shindig/rest/activitystreams
- Endpoint: greenHouse, as stated in the Faces-config.xml
- Give the datasource a variable name, so you can connect to it in your ViewPanel
- Specify in your ViewPanel as value the variable name of the datasource
- And give the ViewPanel also a variable name

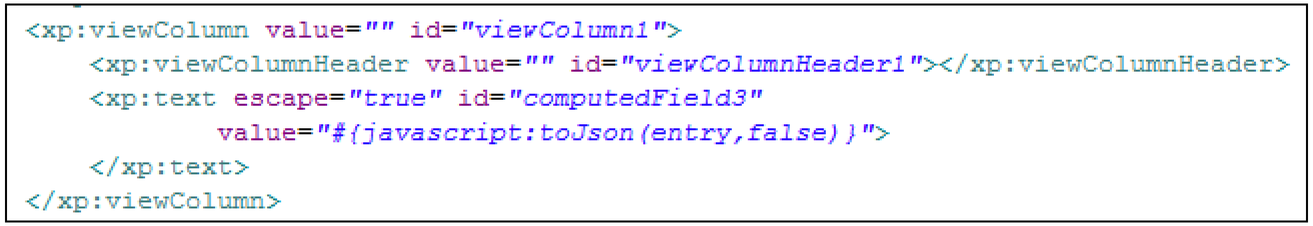


```
<xp:viewPanel rows="30" id="viewPanel1" showColumnHeader="false" var="entry"
    value="activityStreams1" partialRefresh="true">
    <xp:this.data>
        <xe:activityStreamsData
            serviceUrl="/vulcan/shindig/rest/activitystreams"
            endpoint="greenHouse" var="activityStreams1">
        </xe:activityStreamsData>
    </xp:this.data>
```

e-office

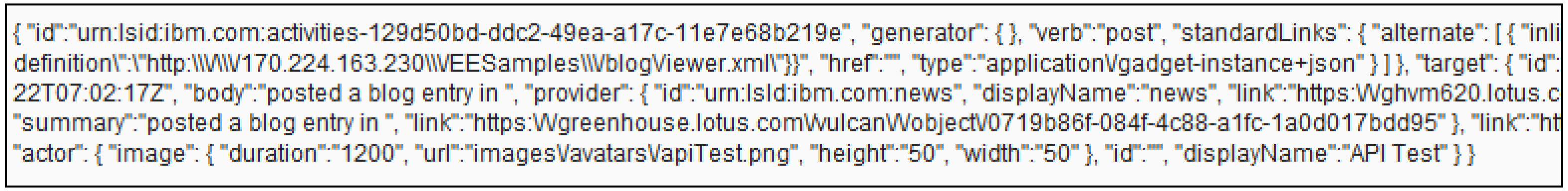
Get the actual data out of the stream

- Add column to the ViewPanel
- Set of the viewcolumn value to “”;
- Add an computed text and add toJson(entry,false)



```
<xp:viewColumn value="" id="viewColumn1">
    <xp:viewColumnHeader value="" id="viewColumnHeader1"></xp:viewColumnHeader>
    <xp:text escape="true" id="computedField3"
            value="#{javascript:toJson(entry,false)}">
    </xp:text>
</xp:viewColumn>
```

- If you preview the Xpage, you will get plain JSON



{ "id":"urn:lsid:ibm.com:activities-129d50bd-ddc2-49ea-a17c-11e7e68b219e", "generator": { }, "verb":"post", "standardLinks": { "alternate": [{ "inli
definition\":\"http:\\/\\/170.224.163.230\\/EESamples\\/blogViewer.xml\"}}", "href":"", "type":"application\/gadget-instance+json" }] }, "target": { "id":
22T07:02:17Z", "body":"posted a blog entry in ", "provider": { "id":"urn:lsld:ibm.com:news", "displayName":"news", "link":"https:\/\/ghvm620.lotus.c
"summary":"posted a blog entry in ", "link":"https:\/\/greenhouse.lotus.com\/vulcan\/object\/0719b86f-084f-4c88-a1fc-1a0d017bdd95" }, "link":"ht
"actor": { "image": { "duration":"1200", "url":"images\/avatars\/apiTest.png", "height":"50", "width":"50" }, "id":"", "displayName":"API Test" } }

- (accessing the Activity Stream for the 1st time, you will need to grant access your application to the Social Business Toolkit)

JSON - some usefull properties

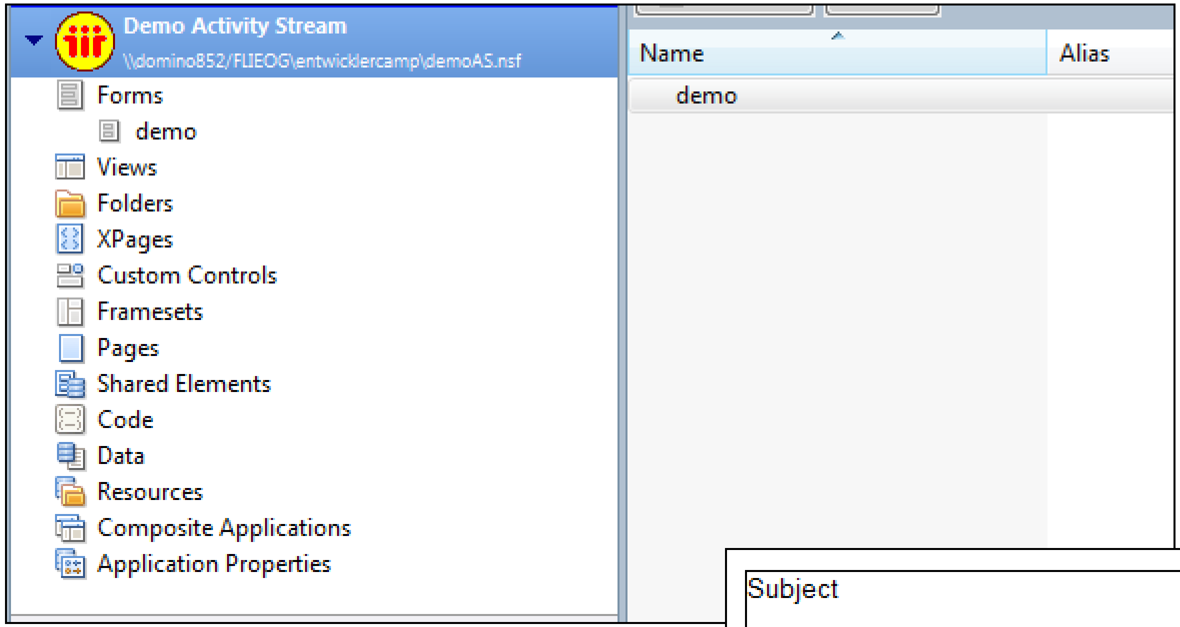
- Title: entry.title
- Posted date: new java.util.Date(parseInt(entry.postedTime)
- ID: entry.id
- Image: entry.actor.image.url
- Body: entry.body
- JSON of the links to the source: entry.standardLinks.alternate[0].inline

Agenda

- Introduction
- Social Business Toolkit
- oAuth and OpenSocial
- Let’s connect to the Activity Stream
- Post to the Activity Stream
- Delete from the Activity Stream
- Q&A
- Usefull links

Post to the Activity Stream

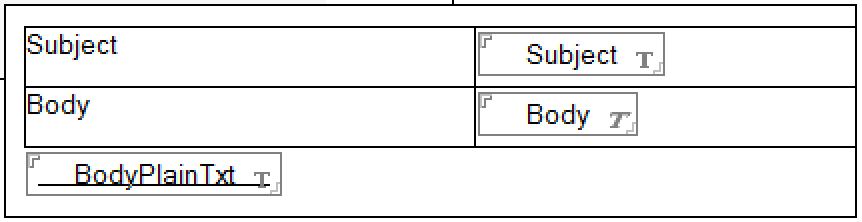
- Create a ‘classic’ Form, with 2 fields



Demo Activity Stream
\\domino852/FLIEOG\entwicklercamp\demoAS.nsf

- Forms
  - demo
- Views
- Folders
- XPages
- Custom Controls
- Framesets
- Pages
- Shared Elements
- Code
- Data
- Resources
- Composite Applications
- Application Properties

Name	Alias
demo	

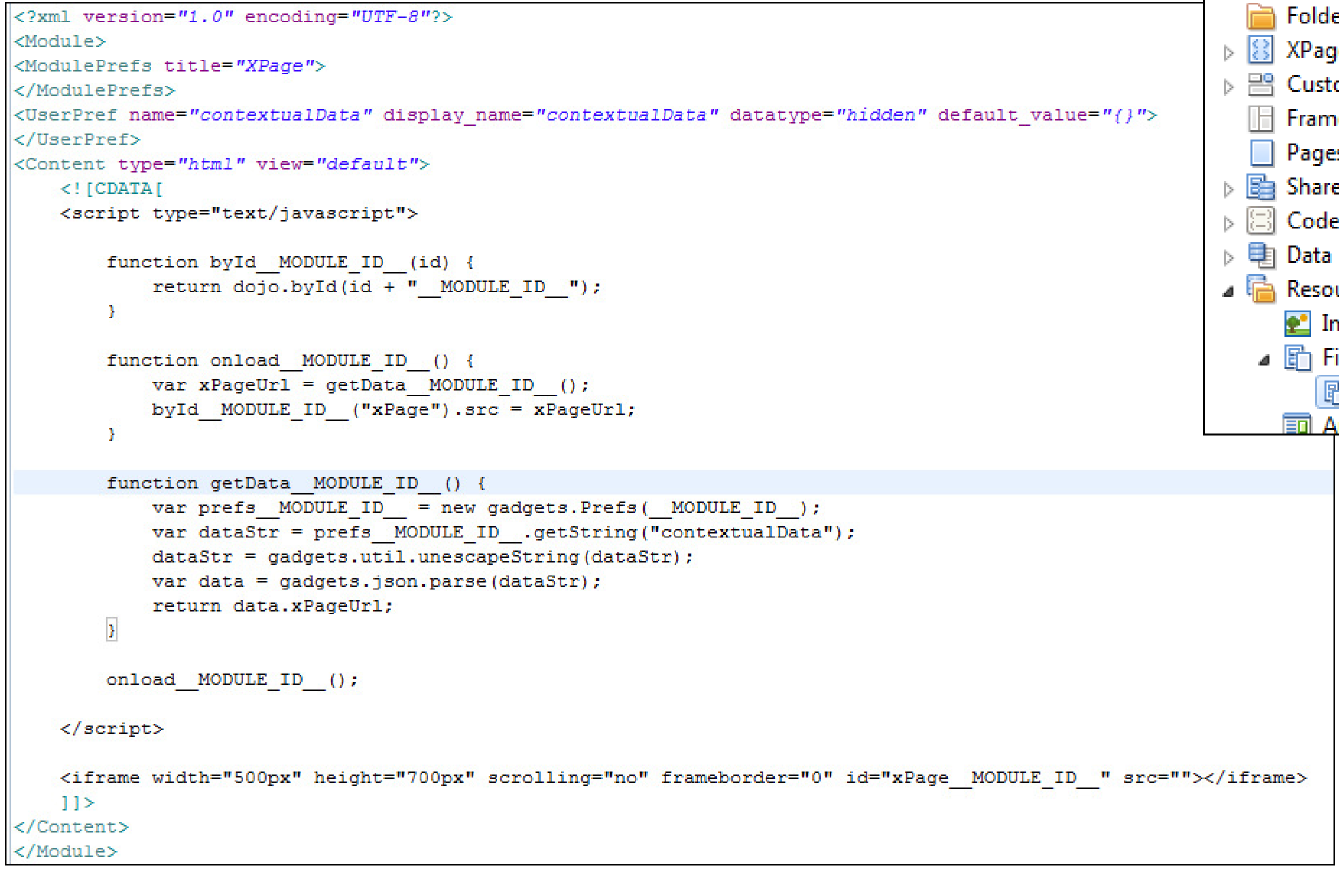


Subject	Subject T
Body	Body T
BodyPlainTxt T	

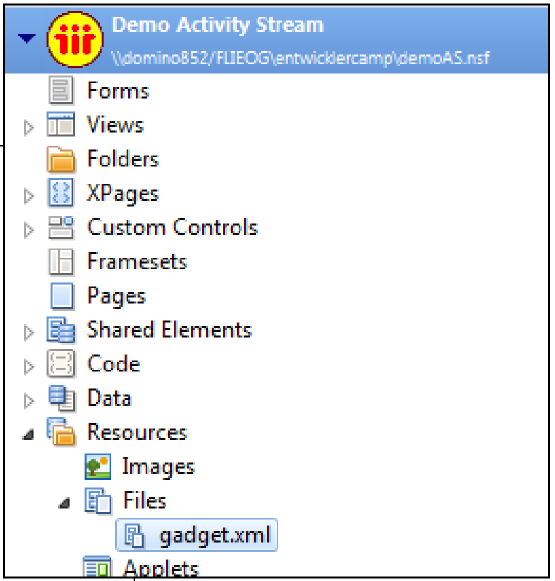
e-office

Post to the Activity Stream

- Create a gadget.xml file in the resources

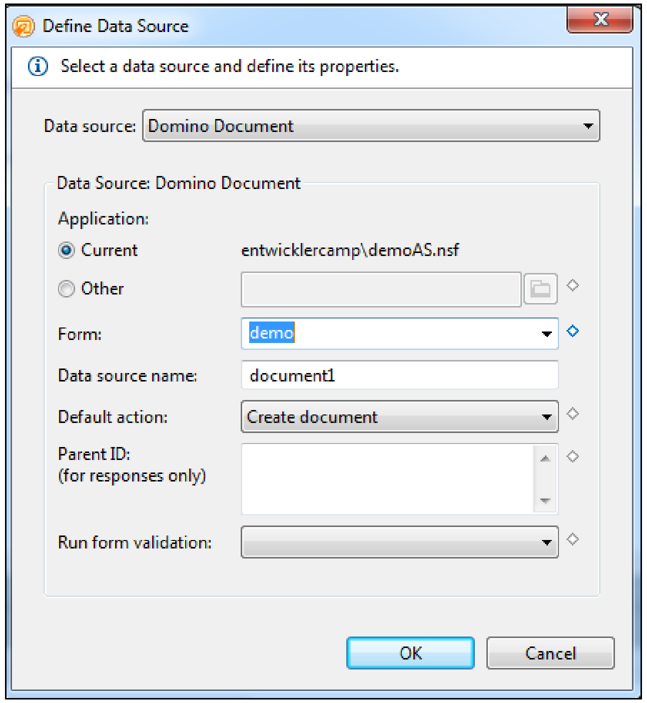


```
<?xml version="1.0" encoding="UTF-8"?>
<Module>
<ModulePrefs title="XPage">
</ModulePrefs>
<UserPref name="contextualData" display_name="contextualData" datatype="hidden" default_value="{}">
</UserPref>
<Content type="html" view="default">
    <![CDATA[
    <script type="text/javascript">

        function byId__MODULE_ID__(id) {
            return dojo.byId(id + "__MODULE_ID__");
        }

        function onload__MODULE_ID__() {
            var xPageUrl = getData__MODULE_ID__();
            byId__MODULE_ID__("xPage").src = xPageUrl;
        }

        function getData__MODULE_ID__() {
            var prefs__MODULE_ID__ = new gadgets.Prefs(__MODULE_ID__);
            var dataStr = prefs__MODULE_ID__.getString("contextualData");
            dataStr = gadgets.util.unescapeString(dataStr);
            var data = gadgets.json.parse(dataStr);
            return data.xPageUrl;
        }

        onload__MODULE_ID__();

    </script>

    <iframe width="500px" height="700px" scrolling="no" frameborder="0" id="xPage__MODULE_ID__" src=""></iframe>
    ]]>
</Content>
</Module>
```



Post to the Activity Stream

- Create XPages and make binding to the Demo form.



Define Data Source

Select a data source and define its properties.

Data source: Domino Document

Data Source: Domino Document

Application:

[✓ Current] entwicklercamp\demoAS.nsf

Other

Form: demo

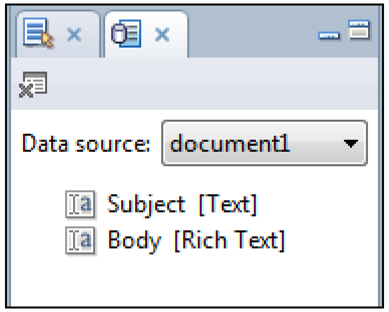
Data source name: document1

Default action: Create document

Parent ID:
(for responses only)

Run form validation:

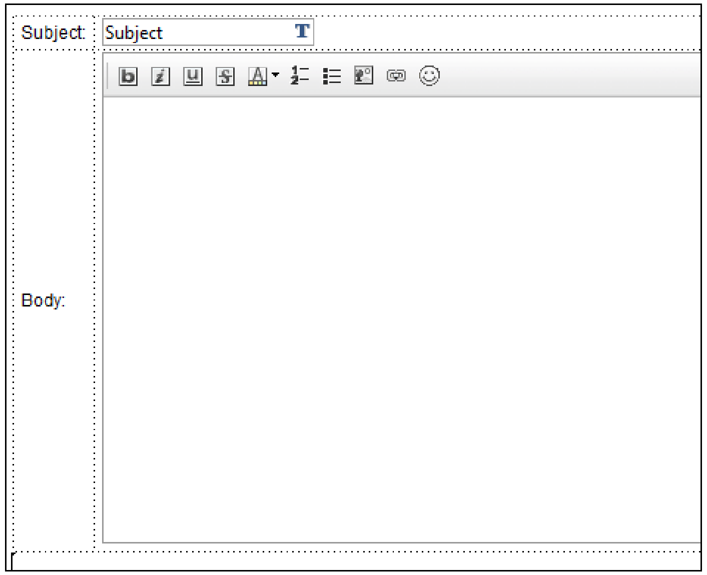
OK Cancel



Data source: document1

Subject [Text]

Body [Rich Text]



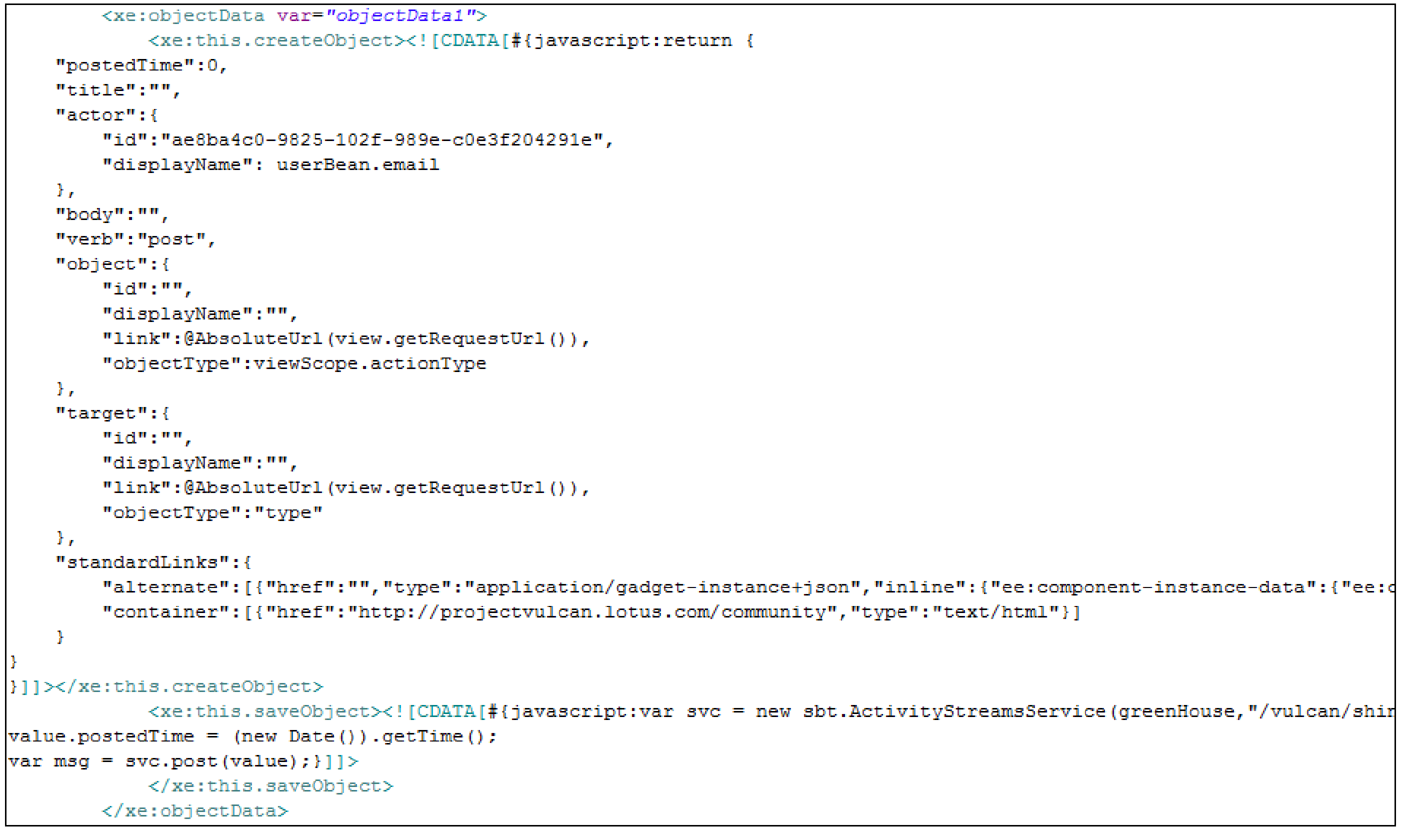
Subject: Subject

Body:

e-office

Post to the Activity Stream

- Add ObjectData control to the XPage
- ObjectData has 2 components
    - ✓ CreateObject: to compose the object JSON
    - ✓ SaveObject: save the object to an url.

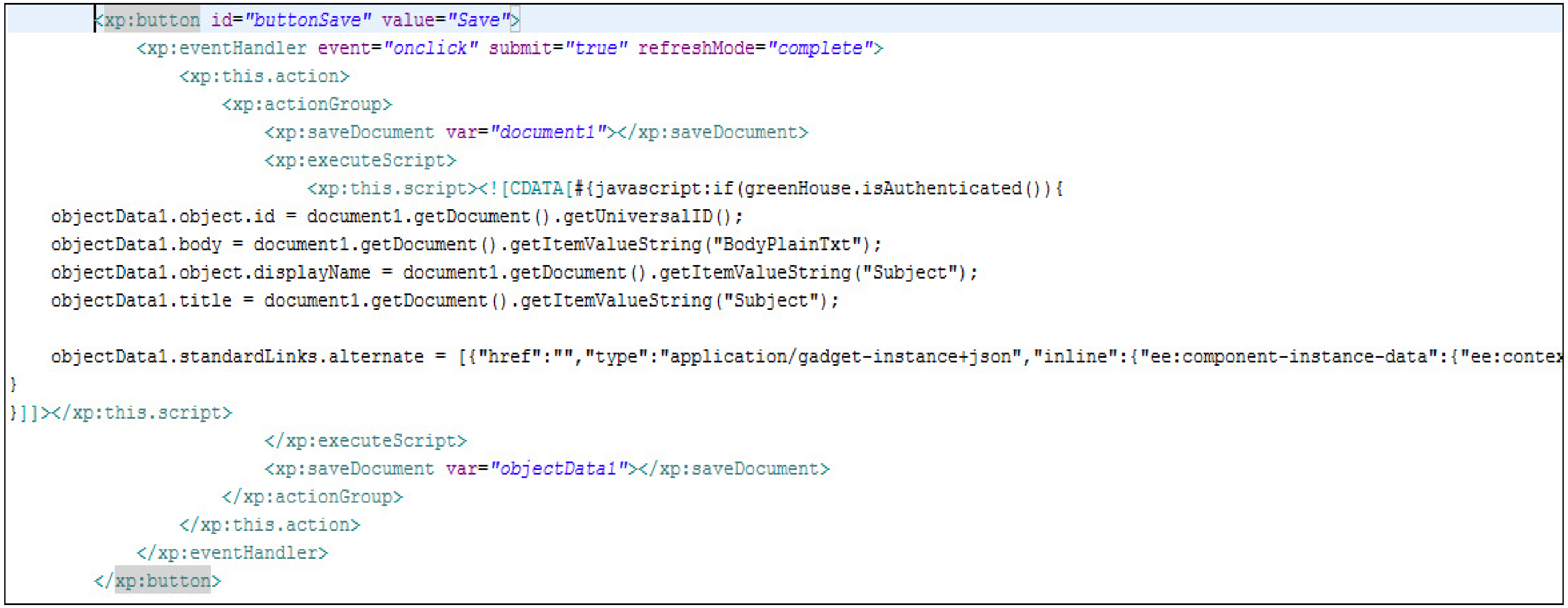


```
        <xe:objectData var="objectData1">
            <xe:this.createObject><![CDATA[#{javascript:return {
    "postedTime":0,
    "title":"",
    "actor":{
        "id":"ae8ba4c0-9825-102f-989e-c0e3f204291e",
        "displayName": userBean.email
    },
    "body":"",
    "verb":"post",
    "object":{
        "id":"",
        "displayName":"",
        "link":@AbsoluteUrl(view.getRequestUrl()),
        "objectType":viewScope.actionType
    },
    "target":{
        "id":"",
        "displayName":"",
        "link":@AbsoluteUrl(view.getRequestUrl()),
        "objectType":"type"
    },
    "standardLinks":{
        "alternate":[{"href":"","type":"application/gadget-instance+json","inline":{"ee:component-instance-data":{"ee:c
        "container":[{"href":"http://projectvulcan.lotus.com/community","type":"text/html"}]
    }
}
}]]></xe:this.createObject>
            <xe:this.saveObject><![CDATA[#{javascript:var svc = new sbt.ActivityStreamsService(greenHouse,"/vulcan/shir
value.postedTime = (new Date()).getTime();
var msg = svc.post(value);}]]>
            </xe:this.saveObject>
        </xe:objectData>
```

e-office

Post to the Activity Stream

- Save the datasources
  - ✓ First save datasource linked to notes document
  - ✓ Collect some field values, e.g. documentUniqueID
  - ✓ Add values to ObjectData
  - ✓ Save datasource linked to ObjectData



```
        <xp:button id="buttonSave" value="Save">
            <xp:eventHandler event="onclick" submit="true" refreshMode="complete">
                <xp:this.action>
                    <xp:actionGroup>
                        <xp:saveDocument var="document1"></xp:saveDocument>
                        <xp:executeScript>
                            <xp:this.script><![CDATA[#{javascript:if(greenHouse.isAuthenticated()){
    objectData1.object.id = document1.getDocument().getUniversalID();
    objectData1.body = document1.getDocument().getItemValueString("BodyPlainTxt");
    objectData1.object.displayName = document1.getDocument().getItemValueString("Subject");
    objectData1.title = document1.getDocument().getItemValueString("Subject");

    objectData1.standardLinks.alternate = [{"href":"","type":"application/gadget-instance+json","inline":{"ee:component-instance-data":{"ee:contex
}
}]]></xp:this.script>
                        </xp:executeScript>
                        <xp:saveDocument var="objectData1"></xp:saveDocument>
                    </xp:actionGroup>
                </xp:this.action>
            </xp:eventHandler>
        </xp:button>
```

e-office

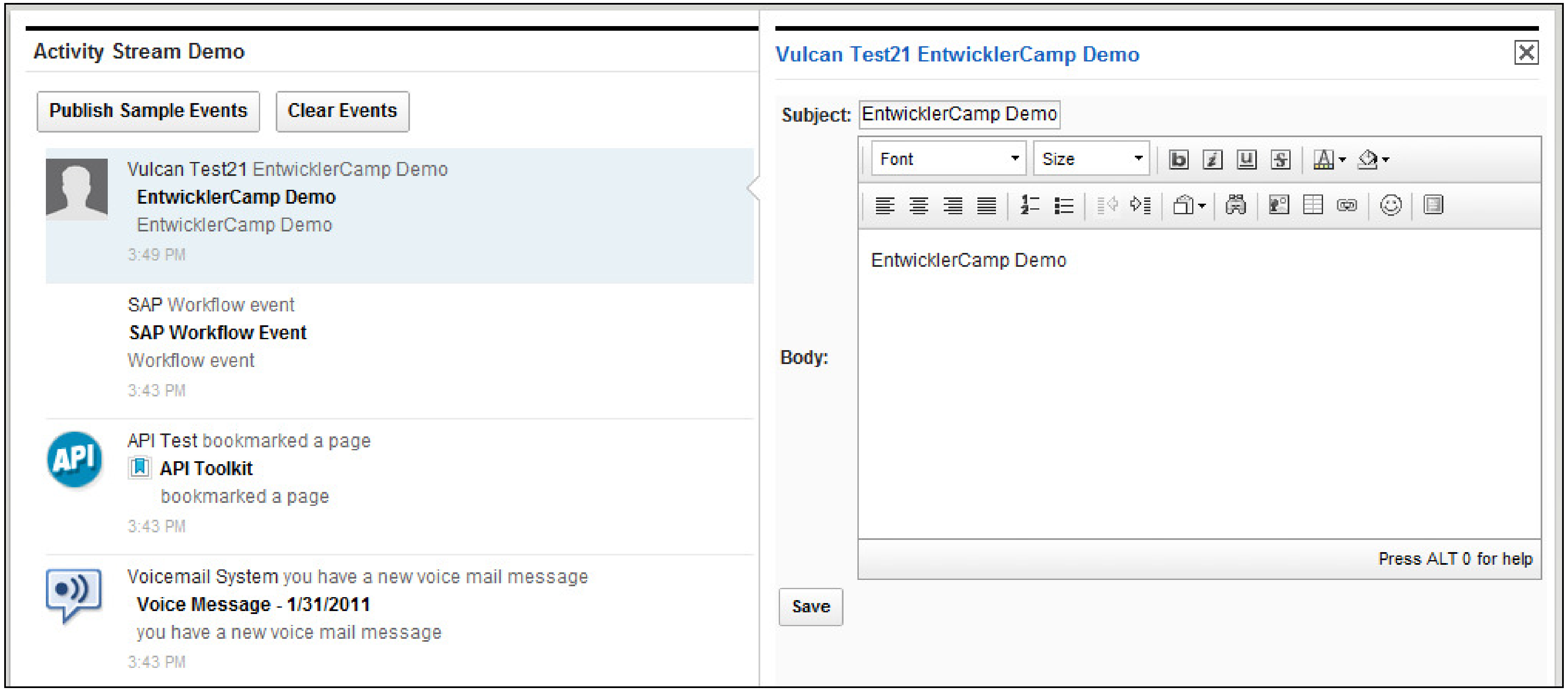
Post to the Activity Stream

- Imported coding to define the OpenSocial gadget syntax
  - ✓ {"ee:context":"{\"bookmarkTitle\":\"EntwicklerCamp2012\",\"xPageUrl\":\"
http:\\\/\\\/ld09.e-
office.com\\\/entwicklercamp//demoAS.nsf\\\/demo.xsp?id="+document1
.getDocument().getUniversalID()+"\"}","ee:component-
definition":"http:\/\/ld09.e-
office.com\/entwicklercamp/demoAS.nsf\/gadget.xml"}

e-office

Post to the Activity Stream

- Refresh Activity Stream
- Click the new entry and you will see the embedded experience


Activity Stream Demo

Vulcan Test21 EntwicklerCamp Demo
EntwicklerCamp Demo
EntwicklerCamp Demo
3:49 PM

SAP Workflow event
SAP Workflow Event
Workflow event
3:43 PM

API Test bookmarked a page
API Toolkit
bookmarked a page
3:43 PM

Voicemail System you have a new voice mail message
Voice Message - 1/31/2011
you have a new voice mail message
3:43 PM

Vulcan Test21 EntwicklerCamp Demo

Subject: EntwicklerCamp Demo

Body: EntwicklerCamp Demo

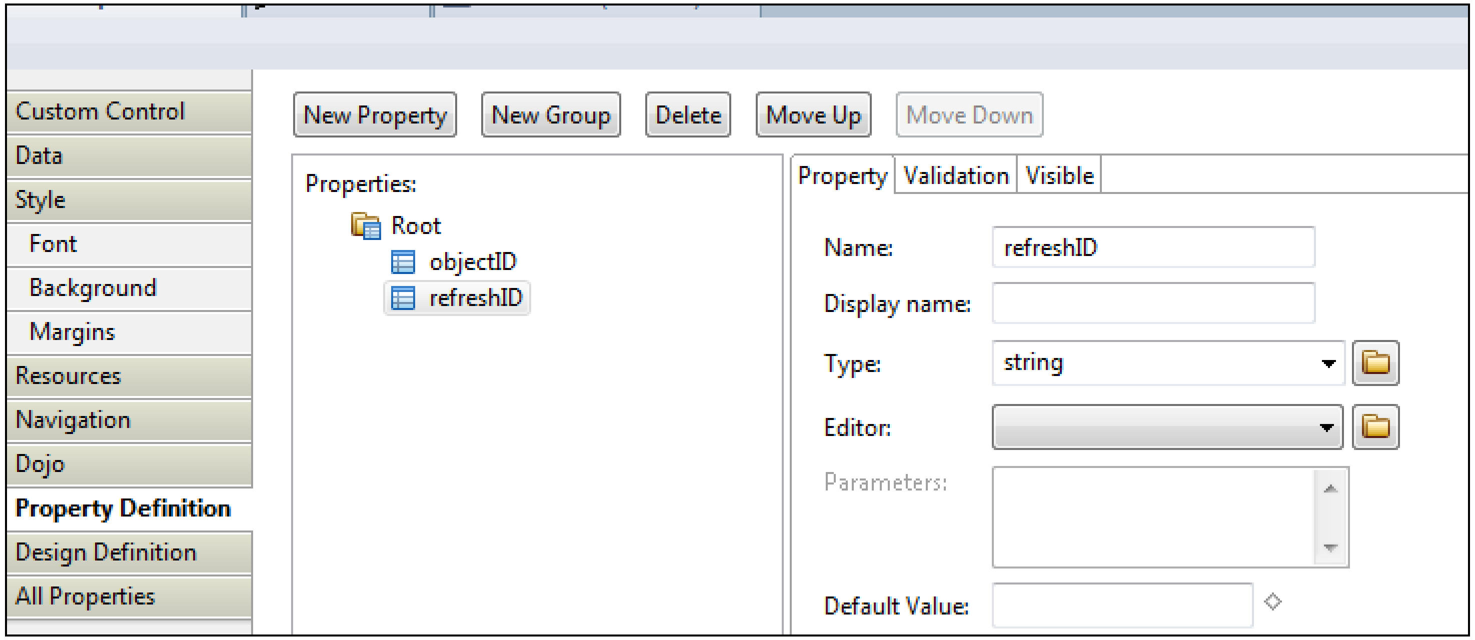
e-office

Agenda

- Introduction
- Social Business
- oAuth and OpenSocial
- Let's connect to the Activity Stream
- Post to the Activity Stream
- **Delete from the Activity Stream**
- Q&A
- Usefull links

Delete from the Activity Stream

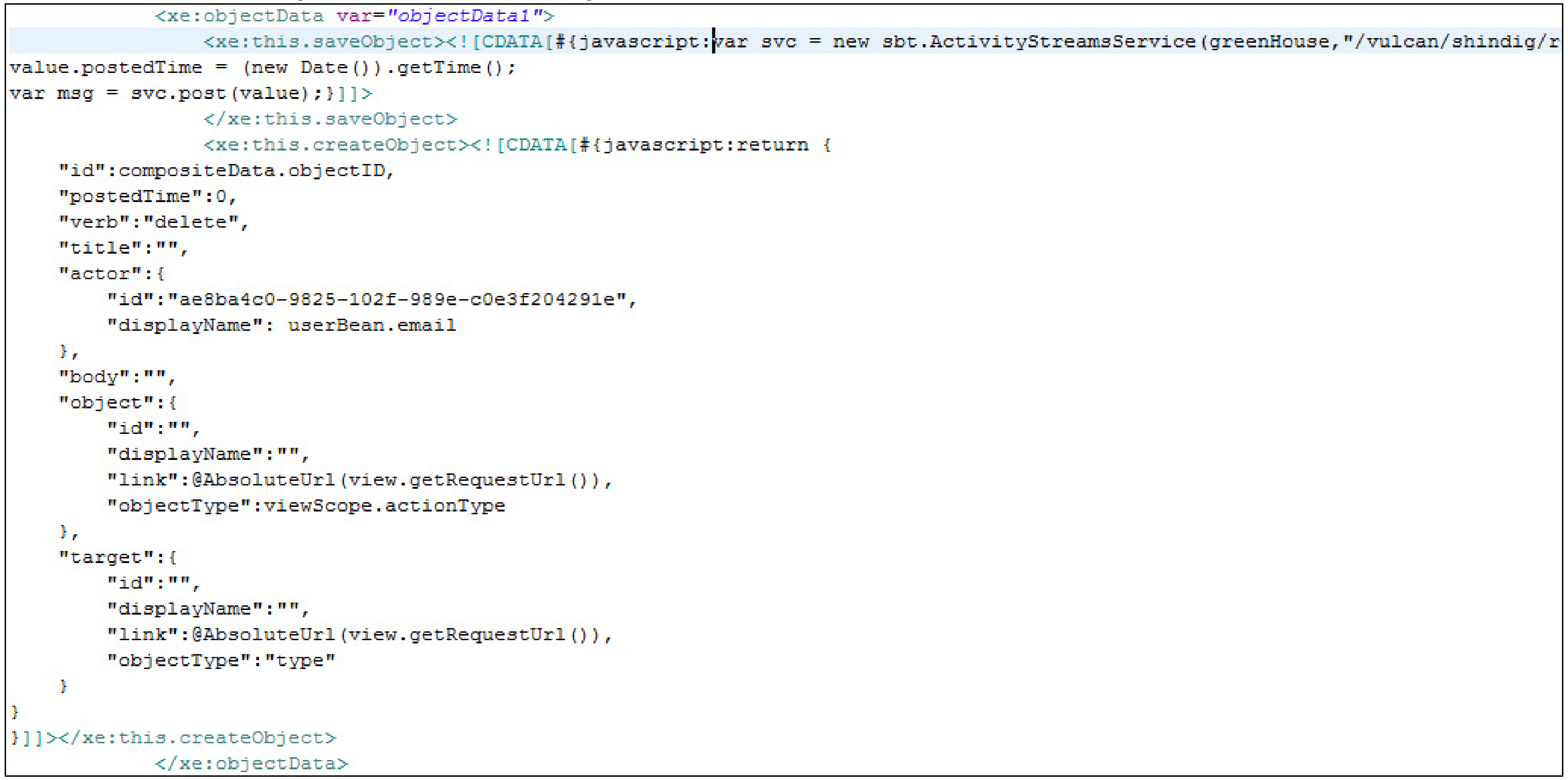
- Create a Custom Control for the Delete Button
- Add custom properties
  - ✓ objectID, identification of entry in ActivityStream
  - ✓ refreshID, component to refresh after delete Action



Name: refreshID
Display name:
Type: string
Editor:
Parameters:
Default Value:

Delete from the Activity Stream

- Add ObjectData control to Custom Control
- ObjectData has 2 components
  - ✓ CreateObject: to compose the object JSON
  - ✓ SaveObject: save the object to an url.



```
            <xe:objectData var="objectData1">
                <xe:this.saveObject><![CDATA[#{javascript:var svc = new sbt.ActivityStreamsService(greenHouse,"/vulcan/shindig/r
value.postedTime = (new Date()).getTime();
var msg = svc.post(value);}]]>
                </xe:this.saveObject>
                <xe:this.createObject><![CDATA[#{javascript:return {
    "id":compositeData.objectID,
    "postedTime":0,
    "verb":"delete",
    "title":"",
    "actor":{
        "id":"ae8ba4c0-9825-102f-989e-c0e3f204291e",
        "displayName": userBean.email
    },
    "body":"",
    "object":{
        "id":"",
        "displayName":"",
        "link":@AbsoluteUrl(view.getRequestUrl()),
        "objectType":viewScope.actionType
    },
    "target":{
        "id":"",
        "displayName":"",
        "link":@AbsoluteUrl(view.getRequestUrl()),
        "objectType":"type"
    }
}
}]]></xe:this.createObject>
            </xe:objectData>
```

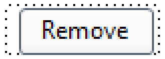
e-office

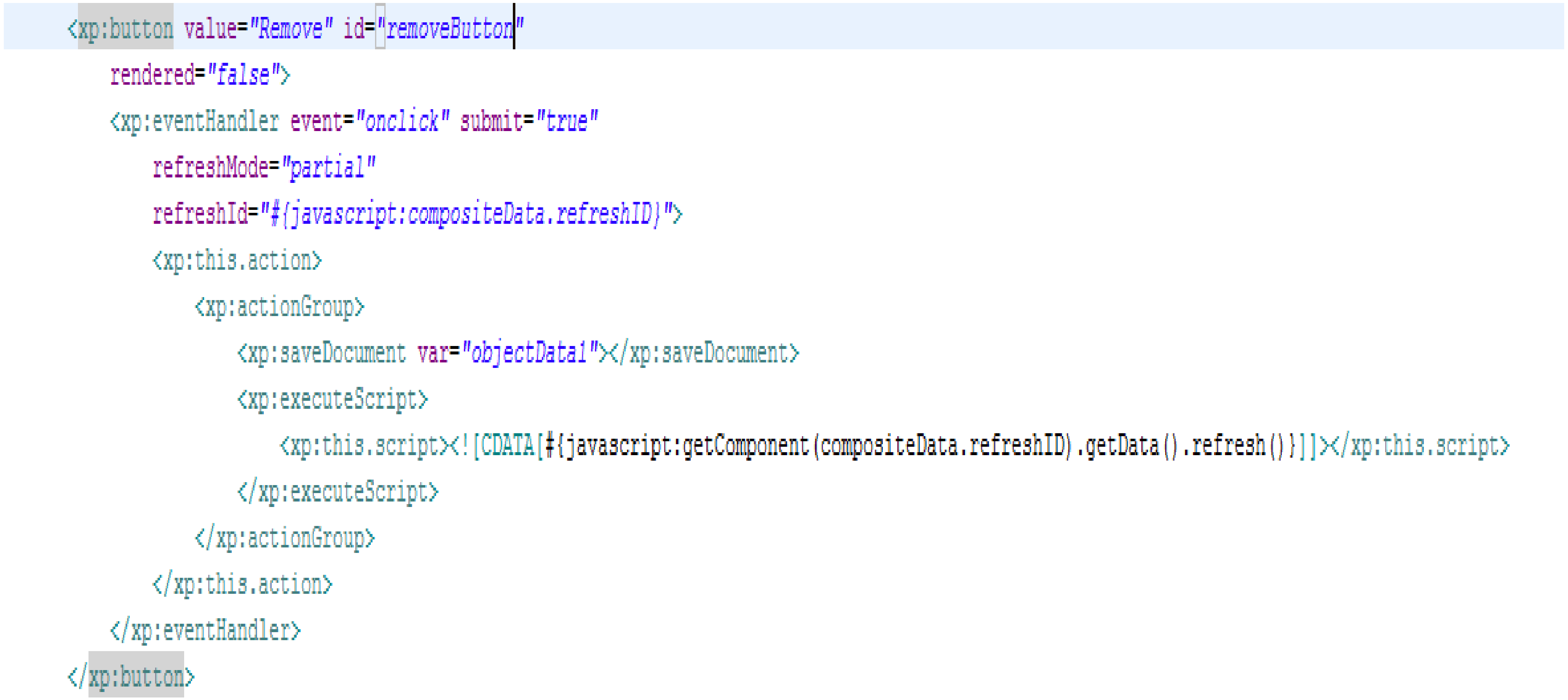
Delete from the Activity Stream

- The service URL: /vulcan/shindig/rest/activitystreams/@me/@all/@all?X-HTTP-Method-Override=DELETE&activityEntryId="+compositeData.objectID
- Endpoint: greenHouse, as stated in the Faces-config.xml

Delete from the Activity Stream

- Add button control to the Custom Control.
- And add onClick event to save datasource and refresh the viewpanel



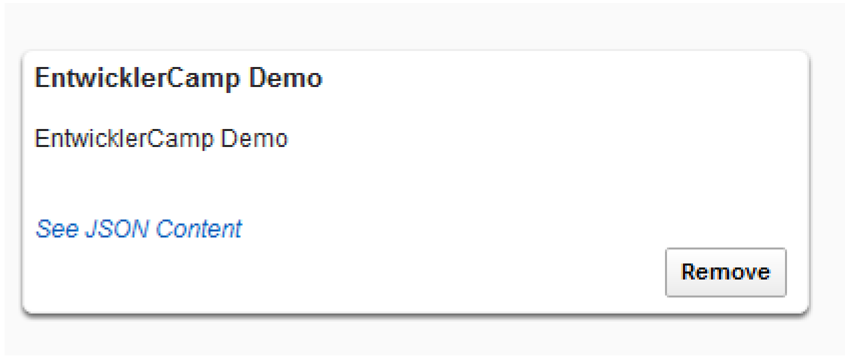


```
<xp:button value="Remove" id="removeButton"
    rendered="false">
    <xp:eventHandler event="onclick" submit="true"
        refreshMode="partial"
        refreshId="#{javascript:compositeData.refreshID}">
        <xp:this.action>
            <xp:actionGroup>
                <xp:saveDocument var="objectData1"></xp:saveDocument>
                <xp:executeScript>
                    <xp:this.script><![CDATA[#{javascript:getComponent(compositeData.refreshID).getData().refresh()}]]></xp:this.script>
                </xp:executeScript>
            </xp:actionGroup>
        </xp:this.action>
    </xp:eventHandler>
</xp:button>
```

e-office

Delete from the Activity Stream

- Go back to the Activity Stream XPage
- Add Delete Action Custom Control to the ViewColum
- Pass some custom properties to identify the entry and for the partial refresh


EntwicklerCamp Demo

EntwicklerCamp Demo

See JSON Content

```
<!-- Delete Button Custom Control -->
<xc:ccDeleteButton
    refreshID="viewPanel1"
    objectID="#{javascript:entry.id}">
</xc:ccDeleteButton>
```

e-office

Q&A

e-office

Usefull links

- OpenNTF Extention Library
  - ✓ http://extlib.openntf.org/
- Social Business Toolkit Activity Stream
  - ✓ https://greenhouse.lotus.com/activitystream/
- API explorer
  - ✓ https://greenhouse.lotus.com/sbtapiexplorer/mainpage.jsp

How to reach me

- Twitter: @flinden68
- Blog: http://www.domino-weblog.nl/
- E-mail: fli@e-office.com