



AD-1387

Outside The Box: Integrating with Non-Domino Apps using XPages and Java

Julian Robichaux, panagenda

Kathy Brown, PSC Group

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

Who Are We?



- Kathy Brown
 - Senior Consultant, PSC Group LLC
 - IBM Champion
 - @IamKathyBrown
 - runningnotes.net



Make Every Moment Count



Who Are We?

- Julian Robichaux
 - Senior Application Developer, panagenda
 - IBM Champion
 - @jrobichaux
 - nsf-tools.com



Make Every Moment Count



Why Are We Here?

- Application integration
 - Get data from other places
 - Put data in other places
- Ideally this will be invisible to the end-user
- We will show you high-level detail (and code!) in this presentation, even more is in the sample database available for download

Make Every Moment Count



Connect2016

The Premier Social Business and Digital Experience Conference



What Will We Be Using?

- IBM Domino® Server 9.0.1 FP4
- IBM Domino Designer (DDE) 9.0.1 FP4
- Java 1.6 (as included with Domino and DDE)
- XPages
- Other bits and bobs as mentioned throughout the presentation

Make Every Moment Count



Bridging the Gap: Converting Data to Java Objects

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Data Basics



- Before we start getting data, let's talk about what the data usually looks like
 - XML
 - JSON
 - CSV
- But Java is made of objects...
- How do we convert from structured data to objects?

Make Every Moment Count

Easy And Old School: CSV



- Comma-separated values
- Simple data format, been used for years, still very common
- Boooorrrriiiinnnnngg
 - yawn
- Stick with us, this will make the other (COOLER) examples easier to understand

Make Every Moment Count

Here's What Our CSV Data Looks Like



Make Every Moment Count

```
"title", "studio", "year", "adjustedRevenue", "unadjustedRevenue"
```

```
"Gone With The Breeze", "NGN", 1939, 1750000000, 200000000
```

```
"Star Battles", "Foz", 1977, 1500000000, 450000000
```

```
"The Sound Of MP3s", "Foz", 1965, 1250000000, 160000000
```

```
"I.T.: The Computer Guys", "Univixen", 1982, 1200000000, 425000000
```

```
"Titantuan", "Parallax", 1997, 1150000000, 650000000
```

And Our Corresponding Java Object



Make Every Moment Count

```
public class MovieBean {  
    private String title;  
    private String studio;  
    private int year;  
    private int adjustedRevenue;  
    private int unadjustedRevenue;  
  
    public MovieBean() {}  
  
    public String getTitle() { return title; }  
    public void setTitle(String title) { this.title = title; }  
    public String getStudio() { return studio; }  
    public void setStudio(String studio) { this.studio = studio; }  
    public int getYear() { return year; }  
    public void setYear(int year) { this.year = year; }  
    public int getAdjustedRevenue() { return adjustedRevenue; }  
    public void setAdjustedRevenue(int adjustedRevenue) { this.adjustedRevenue = adjustedRevenue; }  
    public int getUnadjustedRevenue() { return unadjustedRevenue; }  
    public void setUnadjustedRevenue(int unadjustedRevenue) { this.unadjustedRevenue = unadjustedRevenue; }  
}
```



Connect2016

The Premier Social Business and Digital Experience Conference

Java Beans!



- **Java Beans** are just Java objects with:
 - private fields
 - getter and setter methods to access the fields
 - a “no-arg” constructor
- Perfect for representing structured data elements

```
public class EasyBean {  
    private String name;  
    private int count;  
  
    public EasyBean() {}  
  
    public String getName() { return name; }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public int getCount() { return count; }  
    public void setCount(int count) {  
        this.count = count;  
    }  
}
```

The Bean And Its Data



Make Every Moment Count

```
public class MovieBean {  
    private String title;  
    private String studio;  
    private int year;  
    private int adjustedRevenue;  
    private int unadjustedRevenue;  
  
    public MovieBean() {}  
  
    public String getTitle() { return title; }  
    public void setTitle(String title) { this.title = title; }  
    public String getStudio() { return studio; }  
    public void setStudio(String studio) { this.studio = studio; }  
    public int getYear() { return year; }  
    public void setYear(int year) { this.year = year; }  
    public int getAdjustedRevenue() { return adjustedRevenue; }  
    public void setAdjustedRevenue(int adjustedRevenue) { this.adjustedRevenue = adjustedRevenue; }  
    public int getUnadjustedRevenue() { return unadjustedRevenue; }  
    public void setUnadjustedRevenue(int unadjustedRevenue) { this.unadjustedRevenue = unadjustedRevenue; }  
}
```

```
"title","studio","year","adjustedRevenue","unadjustedRevenue"  
-----  
"Gone With The Breeze","NGN",1939,1750000000,200000000  
-----  
"Star Battles","Foz",1977,1500000000,450000000  
-----  
"The Sound Of MP3s","Foz",1965,1250000000,160000000  
-----  
"I.T.: The Computer Guys","Univixen",1982,1200000000,425000000  
-----  
"Titantuan","Parallax",1997,1150000000,650000000
```



Connect2016

The Premier Social Business and Digital Experience Conference

From Data To Bean



- **Option 1:** Parse the data manually, create objects manually
- **Option 2:** Use a library that parses the data for you into rows and columns, put the row data into objects
- **Option 3:** Use a library that puts the data directly into objects for you!
 - referred to as “unmarshalling” or “deserialization”
 - or “reading” or “parsing”, if that seems confusing

Make Every Moment Count



OpenCSV Library

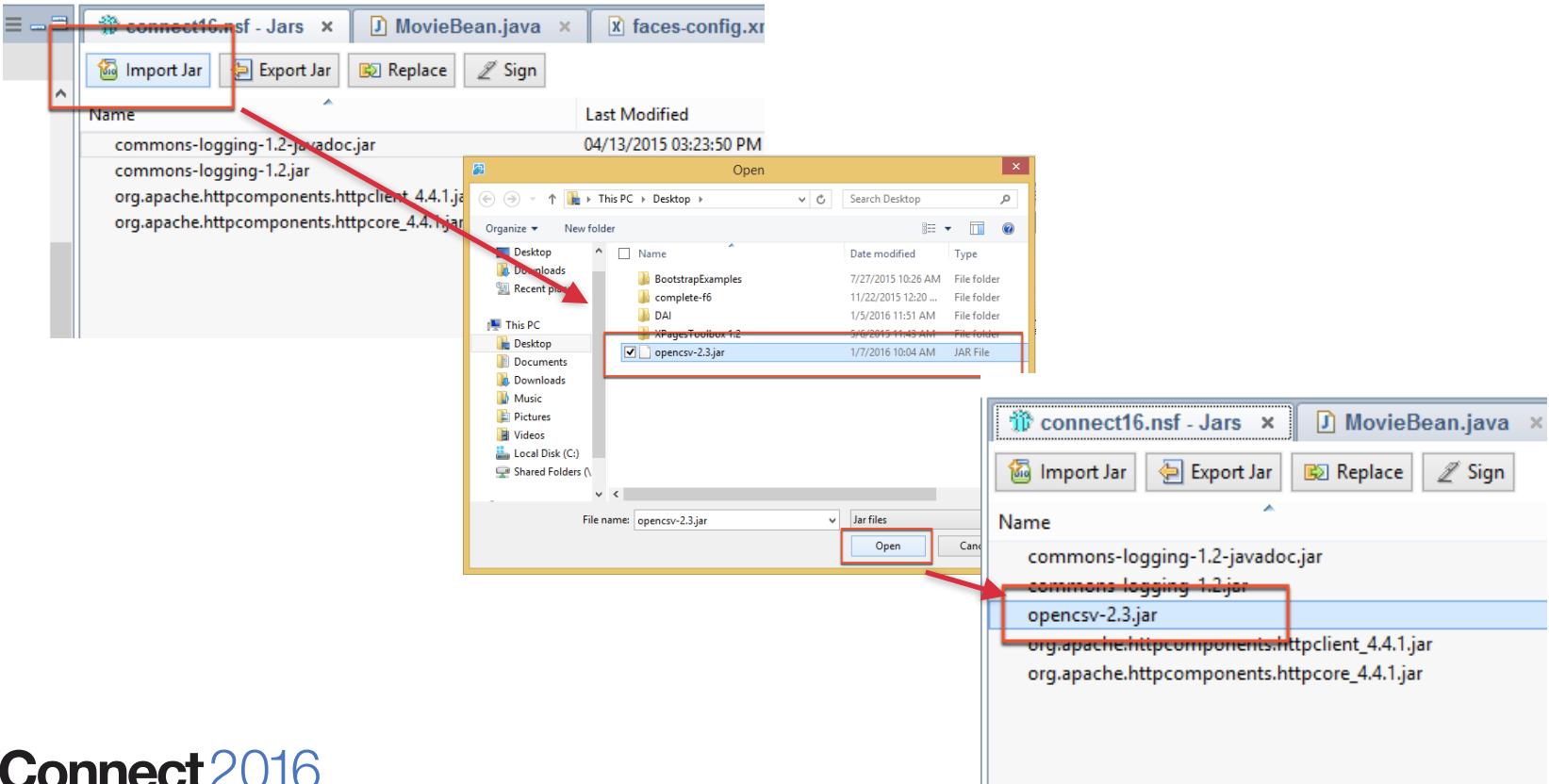
- OpenCSV
 - <http://opencsv.sourceforge.net>
 - Open source, Apache licensed
 - Tiny file size (14k or 42k, depending on version)
- We will use an older version 2.3, because that is compatible with Java 6
 - Latest version 3.6 is compiled for Java 7+
 - `java.lang.UnsupportedClassVersionError: JVMCFRE003 bad major version;`

Make Every Moment Count

Add JAR File to Database



Make Every Moment Count



OpenCSV Parsing Code



Make Every Moment Count

```
public static List convertCSV(String csvText) {  
    try {  
        CSVReader reader = new CSVReader(new StringReader(csvText));  
        ColumnPositionMappingStrategy<MovieBean> strat =  
            new ColumnPositionMappingStrategy<MovieBean>();  
        strat.setType(MovieBean.class);  
        String[] columns = new String[] {  
            "title", "studio", "year", "adjustedRevenue", "unadjustedRevenue"};  
        strat.setColumnMapping(columns);  
  
        CsvToBean csv = new CsvToBean();  
        List list = csv.parse(strat, reader);  
        return list;  
    } catch (Exception e) {  
        // log this!  
        return null;  
    }  
}
```

Awesome! What Did We Learn?



- Use **Java Beans** to hold data
- Use **unmarshalling** to avoid manual effort
- Let **libraries** do the work for you!
 - to a degree... it's always a trade-off

Make Every Moment Count



XML: Manual Parsing

- Java has plenty of XML support built-in
 - DOM, SAX, and STaX for parsing
 - XPath for selecting sets of nodes
- Using these, we can manually read an XML file (or string) and put all the information piece-by-piece into objects
- Or...

Make Every Moment Count



Connect2016

The Premier Social Business and Digital Experience Conference



- Java Architecture for XML Binding (**JAXB**)
- Easily marshal and unmarshal between Java objects and XML
- Included with Java 6+
 - Nothing extra to install for XPages
- Use **Annotations** in the Java Bean source code to indicate which XML elements map to which fields

The XML

```
<moviedata>
  <movie>
    <title>Gone With The Breeze</title>
    <studio>NGN</studio>
    <year>1939</year>
    <adjusted>1750000000</adjusted>
    <unadjusted>200000000</unadjusted>
  </movie>

  <movie>
    <title>Star Battles</title>
    <studio>Foz</studio>
    <year>1977</year>
    <adjusted>1500000000</adjusted>
    <unadjusted>450000000</unadjusted>
  </movie>
```

```
<movie>
  <title>The Sound Of MP3s</title>
  <studio>Foz</studio>
  <year>1965</year>
  <adjusted>1250000000</adjusted>
  <unadjusted>160000000</unadjusted>
</movie>

<movie>
  <title>I.T.: The Computer Guys</title>
  <studio>Univixen</studio>
  <year>1982</year>
  <adjusted>1200000000</adjusted>
  <unadjusted>425000000</unadjusted>
</movie>

<movie>
  <title>Titantuan</title>
  <studio>Parallax</studio>
  <year>1997</year>
  <adjusted>1150000000</adjusted>
  <unadjusted>650000000</unadjusted>
</movie>
</moviedata>
```



Java Bean to Hold Multiple Movies



```
package connect16.example.data;

import java.util.ArrayList;
import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;

@XmlRootElement(name="moviedata") ←
public class MovieBeanList {
    private ArrayList<MovieBean> movies =
        new ArrayList<MovieBean>();

    @XmlElement(name="movie") ←
    public ArrayList<MovieBean> getMovies() {
        return movies;
    }

    public void setMovies(ArrayList<MovieBean> movies) {
        this.movies = movies;
    }
}
```

```
<moviedata>
  <movie>
    <title>Gone With The Breeze</title>
    <studio>NGN</studio>
    <year>1939</year>
    <adjusted>1750000000</adjusted>
    <unadjusted>200000000</unadjusted>
  </movie>
  . .
</moviedata>
```

MovieBean Annotations



```
@XmlElement(name="title")  
public String getTitle() { return title; }  
public void setTitle(String title) {  
    this.title = title; }
```

```
@XmlElement(name="studio")  
public String getStudio() { return studio; }  
public void setStudio(String studio) {  
    this.studio = studio; }
```

```
@XmlElement(name="year")  
public int getYear() { return year; }  
public void setYear(int year) { this.year = year; }
```

```
@XmlElement(name="adjusted")  
public int getAdjustedRevenue() { return adjustedRevenue; }  
public void setAdjustedRevenue(int adjustedRevenue) { this.adjustedRevenue = adjustedRevenue; }
```

```
@XmlElement(name="unadjusted")  
public int getUnadjustedRevenue() { return unadjustedRevenue; }  
public void setUnadjustedRevenue(int unadjustedRevenue) { this.unadjustedRevenue = unadjustedRevenue; }
```

```
<moviedata>  
    <movie>  
        <title>Gone With The Breeze</title>  
        <studio>NGN</studio>  
        <year>1939</year>  
        <adjusted>1750000000</adjusted>  
        <unadjusted>200000000</unadjusted>  
    </movie>  
    . . .  
</moviedata>
```

XML Parsing Code



Make Every Moment Count

```
public static MovieBeanList convertXML(String xml) {  
    try {  
        // see @XML annotations on Java beans  
        JAXBContext jc = JAXBContext.newInstance(MovieBeanList.class);  
        Unmarshaller unmarshaller = jc.createUnmarshaller();  
  
        MovieBeanList movies = (MovieBeanList)unmarshaller.unmarshal(  
            new StringReader(xml));  
        return movies;  
    } catch (Exception e) {  
        // log this!  
        return null;  
    }  
}
```



Connect2016

The Premier Social Business and Digital Experience Conference

And One More Time With JSON



- A JSON parsing library is included with the XPages libraries
 - com.ibm.commons.util.io.json.JsonParser
- It reads JSON into Java Maps and Lists, not custom objects
- Some third-party libraries have unmarshalling support (GSON, FlexJSON) but cause security exceptions
 - You can get around this, but it's not always convenient
- So I wrote my own generic unmarshalling code

Make Every Moment Count

The JSON



Make Every Moment Count

```
[{"movie": {  
    "title": "Gone With The Breeze",  
    "studio": "NGN",  
    "year": 1939,  
    "adjusted": 1750000000,  
    "unadjusted": 200000000  
}},  
 {"movie": {  
    "title": "Star Battles",  
    "studio": "Foz",  
    "year": 1977,  
    "adjusted": 1500000000,  
    "unadjusted": 450000000  
}},  
 {"movie": {  
    "title": "The Sound Of MP3s",  
    "studio": "Foz",  
    "year": 1965,  
    "adjusted": 1250000000,  
    "unadjusted": 160000000  
}},  
 {"movie": {  
    "title": "I.T.: The Computer Guys",  
    "studio": "Univixen",  
    "year": 1982,  
    "adjusted": 1200000000,  
    "unadjusted": 425000000  
}},  
 {"movie": {  
    "title": "Titantuan",  
    "studio": "Parallax",  
    "year": 1997,  
    "adjusted": 1150000000,  
    "unadjusted": 650000000  
}}]  
]
```

Small Change to MovieBean



added **setAdjusted()** and
setUnadjusted() to map to
JSON field names



```
{"movie": {  
    "title": "Gone With The Breeze",  
    "studio": "NGN",  
    "year": 1939,  
    "adjusted": 1750000000,  
    "unadjusted": 200000000  
}}
```

```
@XmlElement(name="adjusted")  
public int getAdjustedRevenue() { return adjustedRevenue; }  
public void setAdjustedRevenue(int adjustedRevenue) { this.adjustedRevenue = adjustedRevenue; }  
public void setAdjusted(int adjustedRevenue) { this.adjustedRevenue = adjustedRevenue; }  
  
@XmlElement(name="unadjusted")  
public int getUnadjustedRevenue() { return unadjustedRevenue; }  
public void setUnadjustedRevenue(int unadjustedRevenue) { this.unadjustedRevenue = unadjustedRevenue; }  
public void setUnadjusted(int unadjustedRevenue) { this.unadjustedRevenue = unadjustedRevenue; }
```

JSON Parsing Code



Make Every Moment Count

```
public static List convertJSON(String json) {  
    try {  
        Object stuff = JsonParser.fromJson(JsonJavaFactory.instance, json);  
        List list = buildObjectFromJSON(MovieBean.class, stuff);  
        return list;  
    } catch (Exception e) {  
        // log this!  
        return null;  
    }  
}
```



Caveats and Tips

- Some code that works for **XPages** will not work for Java **Agents**
 - access to different external libraries
 - default compiler level on agents is still Java 1.4
- Trap (and log) your errors!
 - keep a copy of the “bad” JSON/XML when possible for analysis
- These Java Beans are not quite the same as XPages **Managed Beans** — but you should learn to use those too!

Make Every Moment Count

Bridging the Gap (part deux): Putting Java Objects on an XPage

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect



Make
Every
Moment
Count

So We've Got Objects... Now What?



- How do we display these things?
- The brute force way is by using XAgents or similar:

Make Every Moment Count



```
<?xml version="1.0" encoding="UTF-8"?>
<xp:view
    xmlns:xp="http://www.ibm.com/xsp/core"
    rendered="false">

    <xp:this.afterRenderResponse><![CDATA[#{javascript:
importPackage(connect16.example.data);
var data = "Result: " + Test.convertJSON();

var externalContext = facesContext.getExternalContext();
var writer = facesContext.getResponseWriter();
var response = externalContext.getResponse();
response.setHeader("Cache-Control", "no-cache");
writer.write("<pre>" + data + "</pre>");

writer.endDocument()
}]]></xp:this.afterRenderResponse></xp:view>
```



XPages Data Tables Are An Option

- You can use Java Arrays or Lists as input
- Automatically uses each item in the Array/List as a row in a table
- Customize column values for each row based on item data
- You can even include a pager control

Make Every Moment Count



Connect2016

The Premier Social Business and Digital Experience Conference

Data Table Example

returns an ArrayList of MovieBeans



Make Every Moment Count

```
<?xml version="1.0" encoding="UTF-8"?>
<xp:view xmlns:xp="http://www.ibm.com/xsp/core">
    <xp:dataTable rows="30" id="movieTable" var="movie" style="width:50%">
        <xp:this.value><![CDATA[#{javascript:connect16.example.data.Test.convertJSON();
}]]></xp:this.value>
        <xp:column id="titleColumn">
            <xp:text escape="true" id="titleField" value="#{movie.title}">
            </xp:text>
        </xp:column>
        <xp:column id="studioColumn">
            <xp:text escape="true" id="studioField" value="#{movie.studio}">
            </xp:text>
        </xp:column>
        <xp:column id="yearColumn">
            <xp:text escape="true" id="yearField" value="#{movie.year}">
            </xp:text>
        </xp:column>
    </xp:dataTable>
</xp:view>
```

MovieBean fields
(you can also use
class methods)

Data Table Example



Make Every Moment Count

A screenshot of a web browser window displaying a data table. The URL in the address bar is <http://localhost:1080/MovieData.nsf/DataTableTest.xsp>. The table has three columns: Movie Title, Distributor, and Year.

Gone With The Breeze	NGN	1939
Star Battles	Foz	1977
The Sound Of MP3s	Foz	1965
I.T.: The Computer Guys	Univixen	1982
Titantuan	Paralax	1997



Connect2016

The Premier Social Business and Digital Experience Conference



Repeat Controls and more

Make Every Moment Count

- Using a managed bean allows you to use expression language
 - Expression language allows you to populate:
 - a repeat control
 - input fields
 - really anything in XPages where you can use expression language
 - Pros and cons for using managed beans, check out:
 - <http://www.slideshare.net/RussellMaher/jmp402>

Bootstrapped Repeat on Managed Bean Example



Make Every Moment Count

Connect 2016 Home Ext Lib IBM SocialSDK Quickbase Google Office

AD-1387 Outside The Box: Integrating with Non-Domino Apps using XPages and Java

Label	Studio	Adjusted Gross	Unadjusted Gross	Year
Title	NGN	1750000000	200000000	1939
Gone With The Breeze	Foz	1500000000	450000000	1977
Star Battles				



Make Every Moment Count



The Premier Social Business and Digital Experience Conference



Okay, Let's Go Get Some Data!

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Simple Stuff: Data From A URL



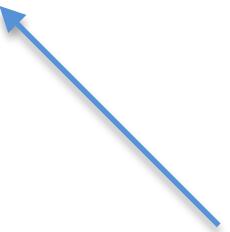
- When we access other servers, we often do it as though we are accessing a URL
 - port 80 and 443 are almost always open for traffic
 - REST services (we'll talk about that next!) usually **are** URL requests
 - a lot of convenient and well-understood security has been built on top of HTTP
- So how hard is it to get data from a URL in Java?

Java URL Fetch: Ridiculously Simple Example



Make Every Moment Count

```
URL url = new URL("http://server/data.xml");
InputStream in = url.openStream();
```



something has to read the stream,
but that's all there is (in a basic sense)



What's An InputStream?

- An InputStream is a “stream” of bytes
- Often sent in chunks, so you don’t have to have the entire contents of the stream in memory unless you want to
- Commonly accepted and understood by Java methods

Make Every Moment Count



The Premier Social Business and Digital Experience Conference

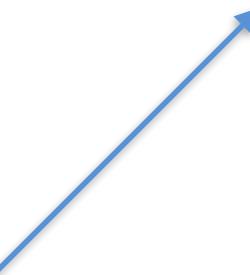
XML As An InputStream



Make Every Moment Count

```
URL url = new URL("https://server/data.xml");
InputStream in = url.openStream();

JAXBContext jc = JAXBContext.newInstance(MovieBeanList.class);
Unmarshaller unmarshaller = jc.createUnmarshaller();
MovieBeanList movies = (MovieBeanList)unmarshaller.unmarshal(in);
```



Converting Bytes To Strings



- What if your Java method needs a String?
 - or a Reader
- Use an InputStreamReader... BUT
- Mind your character sets!
 - Java uses the default system character set by default
 - According to spec, JSON is supposed to be sent as Unicode (usually UTF-8): <https://tools.ietf.org/html/rfc4627>

Make Every Moment Count

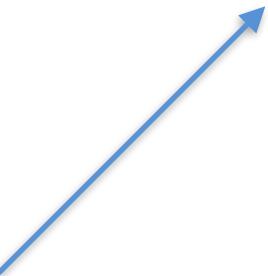
JSON As A Reader



Make Every Moment Count

```
URL url = new URL("https://server/data.js");
InputStream in = url.openStream();

BufferedReader reader = new BufferedReader(
    new InputStreamReader( url.openStream(), "UTF-8" ));
Object stuff = JsonParser.fromJson(JsonJavaFactory.instance, reader);
```

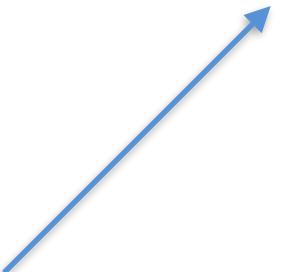


CSV As A Reader



```
URL url = new URL("https://server/data.csv");
InputStream in = url.openStream();

BufferedReader reader = new BufferedReader(
    new InputStreamReader( url.openStream(), "UTF-8" ));
CSVReader csv = new CSVReader(reader);
```



If You Really Really Want A String



Make Every Moment Count

```
URL url = new URL("https://server/data.txt");
BufferedReader reader = new BufferedReader(
    new InputStreamReader( url.openStream(), "UTF-8" ));

StringBuilder sb = new StringBuilder();
char[] buffer = new char[1024];
int size = 0;
while ( (size = reader.read(buffer)) > 0 ) {
    sb.append(buffer, 0, size);
}

reader.close();
String stuff = sb.toString();
```

Is That All You Have To Do?



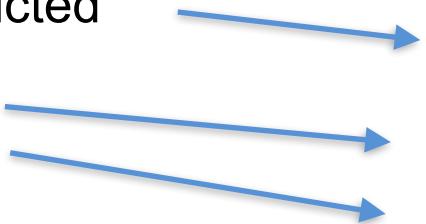
- Well...
 - proxy servers
 - authentication
 - SSL certificates
 - POST requests and headers
- Examples of all those things in the sample database
- When you use other libraries (like the SocialSDK, coming up!) these things are often taken care of for you

Domino Security For Network (and File) Access



- You must have the following permissions in the Security section of the Server doc to access the network:

- EITHER: run unrestricted
- OR: sign XPages
AND run restricted



Programmability Restrictions	Who can -
Sign or run unrestricted methods and operations:	ThoseWhoCanRunUnrestricted
Sign agents to run on behalf of someone else:	ThoseWhoCanSignForOthers
Sign agents or XPages to run on behalf of the invoker:	ThoseWhoCanSignForInvoking
Sign or run restricted LotusScript/Java agents:	ThoseWhoCanRunRestricted

- This is the signature of the XPage AND all associated resources
 - referenced controls, JS libraries, Java code, attached JAR files
 - if any one of those things doesn't have access, the XPage will fail
 - HTTP Web Server: You are forbidden to perform this operation**



Connect2016

The Premier Social Business and Digital Experience Conference

REST Service Example



- REST = Representational State Transfer
- There are a lot of finer points to a proper REST service, but for the purpose of our examples it will be:
 - using a URL + HTTP to get information
 - using a URL + HTTP to send information

Make Every Moment Count

Simple REST Service Example



Make Every Moment Count

```
public static MovieBeanList getMoviesFromREST()
    throws IOException, JAXBException {
String restURL = "http://server/movieService.php?count=5&output=xml";
URL url = new URL(restURL);
InputStream in = url.openStream();

JAXBContext jc = JAXBContext.newInstance(MovieBeanList.class);
Unmarshaller unmarshaller = jc.createUnmarshaller();
return (MovieBeanList)unmarshaller.unmarshal( in );
}
```



Connect2016

The Premier Social Business and Digital Experience Conference

Simple REST Service Example



Make Every Moment Count

```
public static void createDocs(Database db) {  
    try {  
        MovieBeanList movies = getMoviesFromREST();  
        for (MovieBean movie : movies.getMovies()) {  
            Document doc = db.createDocument();  
            doc.replaceItemValue("Form", "MovieData");  
            doc.replaceItemValue("Title", movie.getTitle());  
            doc.replaceItemValue("Studio", movie.getStudio());  
            doc.replaceItemValue("Year", movie.getYear());  
            doc.save();  
            doc.recycle();  
        }  
    } catch (Exception e) {  
        // log this!  
    }  
}
```



Connect2016

The Premier Social Business and Digital Experience Conference



Make Every Moment Count



The Premier Social Business and Digital Experience Conference



IBM SocialSDK

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

IBM SocialSDK (aka Social Business Toolkit SDK)



- What is it?
 - A toolkit, provided by IBM, to integrate with Connections, Connections Cloud, Dropbox, Facebook, Twitter, and more
 - Originally part of the XPages Extension Library, split off to its own project in 2013
- Download from OpenNTF or Github
 - <https://openntf.org/p/Social+Business+Toolkit+SDK>
 - <https://github.com/OpenNTF/SocialSDK/releases>

Make Every Moment Count

IBM SocialSDK (aka Social Business Toolkit SDK)



- What's included?
 - Toolkit files
 - OAuth* code
 - Examples
 - Sample XPages application
 - Also has components for Websphere and Tomcat

Make Every Moment Count



*OAuth

- A quick word about OAuth
 - Many APIs require OAuth
 - OAuth authentication is handled in the Social Business Toolkit
 - BUT it's still a really really good thing to understand
 - <http://www.slideshare.net/dominopoint/dd12-oauth-for-domino-developers>

Make Every Moment Count



IBM SocialSDK Install

- How to install
 - IBM Getting Started - https://www-10.lotus.com/ldd/lcwiki.nsf/dx/Installing_on_Domino_Server_SDK1.0
 - using an UpdateSite database
 - doesn't load? See <http://johnjardin.ukuvuma.co.za/2012/05/04/xpages-tip-what-to-check-when-domino-doesnt-install-osgi-plugins-from-updatesite>
 - Circular reference! <http://xomino.com/2015/03/08/ibmsbt-in-xpages-getting-set-up-with-a-social-business-toolkit-part-1/>
 - using file copy technique
 - method may not be “best practice”, but it will get you up and running

Make Every Moment Count

IBM SocialSDK (Quick Install Version)



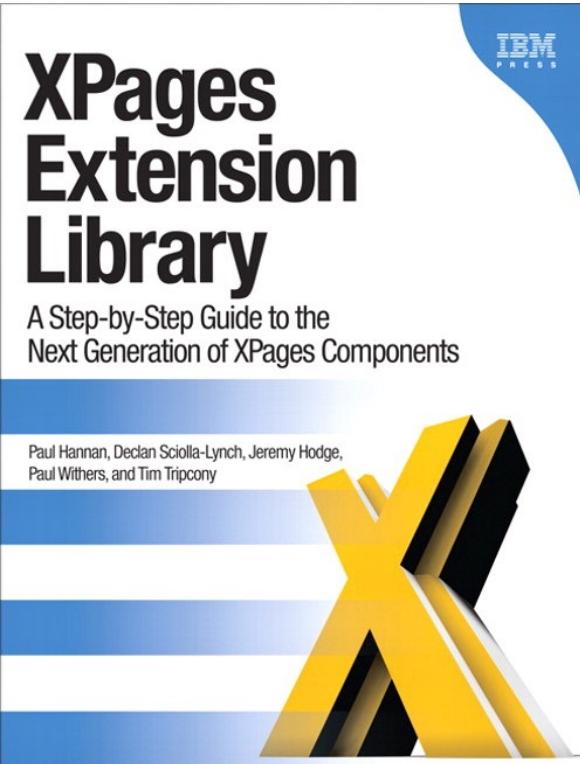
Make Every Moment Count

- Unzip the file from OpenNTF
- Install the plugin in Designer
- Set up an update site OR find the “Features” and “Plugins” folders in the archive and place on your server
 - \data\domino\workspace\applications\eclipse
- Re-start server
- Register your app for OAuth (if needed)
- Open the XPagesSBT.nsf to check that the plugin is working
 - May need to update the endpoints
 - <http://xomino.com/2015/03/10/ibmsbt-in-xpages-getting-set-up-with-a-social-business-toolkit-part-2/>

IBM SocialSDK - What Can You Do?



- Activities
- Bookmarks
- Communities
- Files
- Forums
- Profiles
- and more



old but still
useful
(chapter 13)



Make Every Moment Count



Use the API

- Javadocs:
 - http://infolib.lotus.com/resources/social_business_toolkit/javadoc
 - Source code: <https://github.com/OpenNTF/SocialSDK>
- Examples:
 - XPagesSBT.nsf
 - <https://greenhouse.lotus.com/sbt/sbtplayground.nsf/JavaScriptSnippets.xsp>
 - <https://greenhouse.lotus.com/sbt/sbtplayground.nsf/JavaSnippets.xsp>
 - search for “niklas heidloff sbt demo”



Adding Connections Functionality to Your DB

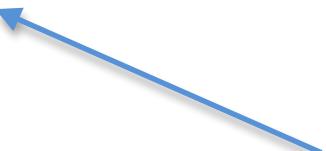
- Copy the following design elements from XPagesSBT.nsf to your database:
 - `_BasicLogin` (XPage)
 - `sbtLoginPagePanel` (Custom Control)
 - `sbtLoginPage` (Custom Control)
- Open `faces-config.xml` in XPagesSBT.nsf
 - copy the `<managed-bean>` info under "Greenhouse Connections" and add it to the `face-config.xml` file in your database
 - modify the "url" and "authenticationPage" references to match your Connections server and local database
- In the **Xsp Properties** of your database (under Application Configuration), make sure **com.ibm.xsp.sbtSDK.library** is checked on the "Page Generation" tab
 - if that's not an option, you didn't install the SocialSDK plugins yet

IBM SocialSDK - My Files Java



Make Every Moment Count

```
public static EntityList<File> getConnectionsFiles() {  
    try {  
        FileService service = new FileService();  
        // API CHANGE! Used to return fileList  
        EntityList<File> myFiles = service.getMyFiles();  
        return myFiles;  
    } catch (Exception e) {  
        // log this!  
        return null;  
    }  
}
```



use this in a dataTable
like we did before

IBM SocialSDK - My Files JavaScript



Make Every Moment Count

```
var fileService = new FileService();
fileService.getMyFiles().then(
    function(files) {
        if (files.length == 0) {
            dom.setText("filesTable", "You do not own any files, you loser.");
        } else {
            for(var i=0; i<files.length; i++){
                var file = files[i];
                createItem(file);
            }
        }
    }
}
```

The screenshot shows the 'My Files' section of the IBM SocialSDK interface. The left sidebar includes links for 'Upload Files', 'New...', 'Pinned Files', 'My Files' (which is selected), 'Shared With Me', 'Shared By Me', 'Sync', 'Community Files', 'PSC Group Files', and 'Trash'. The main area displays a list of files under 'My Files' with the heading 'Files that you own.' It includes columns for file name, owner, creation date, and file preview. Buttons for 'Download', 'Share', 'Add to Sync', 'Add Tags', and 'Move to Trash' are at the top. A 'Display' button with grid and list icons is on the right. At the bottom, there are links for 'Show 10 | 25 | 50 items per page' and 'Feed for these Files | Download All Files'. A message at the bottom states 'You have 1.024 GB of free space remaining (1.024 GB limit)'.

The screenshot shows a web browser window with the URL 'http://www.r-exsp#home'. The title bar says 'Test Document.txt - File'. The page content is titled 'Connections Files' and shows a 'My Files' section. It lists several files with their names, owners, creation dates, and file preview icons. To the right of each file, there are social sharing icons for likes (1), comments (0), and shares (0). A red arrow points from the 'More' button next to the 'Test Document.txt' entry in the screenshot on the left to the social sharing icons for that file in the screenshot on the right. The right screenshot also shows a message at the bottom: 'You have 1.024 GB of free space remaining (1.024 GB limit)'.



Connecting to Other Servers

- The StackExchange example in XPagesSBT.nsf is a good place to start if you want to connect to a custom server
 - stackoverflow managed bean and GenericStackoverflow.xsp XPage
 - <https://api.stackexchange.com/docs/questions>

```
<xe:this.data>
    <xe:restJsonData var='rest' endpoint="stackoverflow" serviceUrl="/2.0/questions"
paramCount="pagesize" paramFirst="page" paramFirstType="page1" splitPath="items">
        <xe:this.urlParameters>
            <xe:urlParameter name="site" value="stackoverflow"></xe:urlParameter>
            <xe:urlParameter name="order" value="desc"></xe:urlParameter>
            <xe:urlParameter name="sort" value="activity"></xe:urlParameter>
        </xe:this.urlParameters>
    </xe:restJsonData>
</xe:this.data>
```



Connecting to Other Servers

- You can also connect directly using a ClientService
 - basic authentication is easy, OAuth takes more work
 - uses Apache commons HttpClient under the covers
 - methods for get, post, put, delete

Make Every Moment Count

```
RestClient client = new RestClient("http://your.rest.server");
client.setAuthenticator(new RestClient.BasicAuthenticator("username", "password"));
Response response = client.get("your/rest/endpoint", ClientService.FORMAT_TEXT);
return (String)response.getData();
```





Make Every Moment Count



The Premier Social Business and Digital Experience Conference



Quickbase

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Quickbase



- <http://www.quickbase.com/api-guide/index.html>
- What You'll Need:
 - An understanding of Quickbase
 - Quickbase account
 - JavaScript and HTML for “simple” actions like embedding a table
 - REST and Java for more complex API calls

Make Every Moment Count

Quickbase



- Embed a Quickbase table in XPages:

Somewhere in head of page:

```
<script lang="javascript" src="https://psc.quickbase.com/db/YOUR_DB_ID?  
a=API_GenResultsTable&appToken=YOUR_APP_TOKEN&jht=1">  
</script>
```

Somewhere in body of page:

```
<script lang="javascript">  
 qdbWrite();  
</script>
```

The screenshot shows a web browser window with three tabs open. The active tab displays an XPage titled "AD-1387 Outside The Box: Integrating with Non-Domino Apps using XPages and Java". Below the title, there is a table with the following data:

Title	Studio	Adjusted Gross	Unadjusted Gross	Year
Gone With The Breeze	NGN	1750000000	200000000	1939
Star Battles	Foz	1500000000	450000000	1977
The Sound Of MP3s	Foz	1250000000	150000000	1965
I.T.: The Computer Guys	Univixen	1200000000	425000000	1982
Tilantuan	Parallax	1150000000	650000000	1997
The Ten Command Lines	Parallax	1125000000	65000000	1956
Jowls	Univixen	1100000000	260000000	1975
Doctor Zorro	NGN	1000000000	110000000	1965
The Exerciser	Warmer Toes	950000000	230000000	1973
Snow Crash And The Seven Chores	Dizzy	940000000	185000000	1937
101 Damnations	Dizzy	860000000	145000000	1961
The Umpire Strikes Out	Foz	845000000	290000000	1980
Ben-Herd	NGN	840000000	74000000	1959
AvAtari	Foz	830000000	760000000	2009



- More complex Quickbase API calls require post and response from Quickbase and your application
- Most calls require authentication
 - XPages application posts username and password to QB
 - QB responds with an authentication token to be used throughout session

Quickbase - Steps for an API call



Make Every Moment Count

```
public static String getResponse(String myURL, String nodePath) {  
    try {  
        URL url = new URL(myURL);  
        DocumentBuilderFactory factory =  
            DocumentBuilderFactory.newInstance();  
        factory.setValidating(false);  
  
        DocumentBuilder builder = factory.newDocumentBuilder();  
        org.w3c.dom.Document dom = builder.parse( url.openStream() );  
        Node node = XPathAPI.selectSingleNode(dom, nodePath);  
        return node.getNodeValue();  
    } catch (Exception e) {  
        // log this!  
        return null;  
    }  
}
```

https://target_domain/db/main?
a=API_Authenticate&username=PTBa
rnum&password=TopSecret&hours=24

<?xml version="1.0" ?>
<qdbapi><action>api_authenticate</action>
<errcode>0</errcode><errtext>No error</
errtext>
<ticket>2_beeinrxmv_dpvx_b_crf8ttndjwyf9bui94
rhciirqcs</ticket>
<userid>112245.efy7</userid></qdbapi>



Make Every Moment Count



The Premier Social Business and Digital Experience Conference



Relational Databases

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

XPages Extension Library



- The XPages Extension Library (ExtLib) began as an OpenNTF project in 2011
 - <https://extlib.openntf.org>
- As of Domino 9.0, basic components of ExtLib are installed with the server by default
 - com.ibm.xsp.extlib.* in domino/osgi/shared/eclipse/plugins
- **However**, the good stuff is still on OpenNTF

Make Every Moment Count



Installing ExtLib

- Super high-level install instructions
 - Download from <https://extlib.openntf.org>
 - Unzip and extract **updateSiteOpenNTF-designer.zip** and **updateSiteOpenNTF.zip**
 - Create 2 different update sites: one for Designer, one for Domino server
 - Install appropriate update sites to server (via notes.ini) and DDE (File - Application - Install)
 - Restart Domino and DDE

https://www-10.lotus.com/ldd/ddwiki.nsf/dx/XPages_Extension_Library_Deployment

Relational Database Support

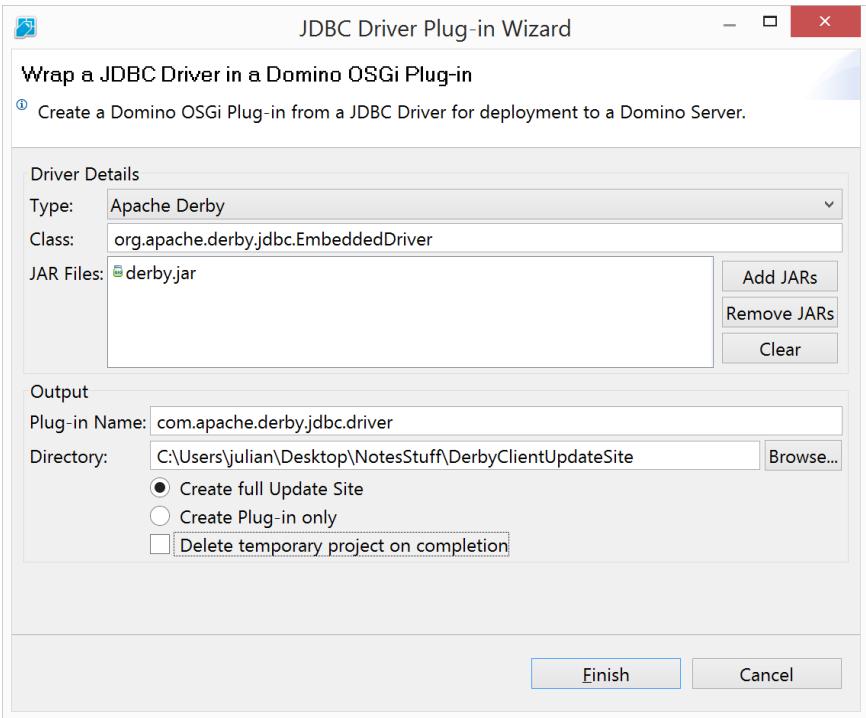


- In 2014, the ExtLib component that allows you to connect XPages directly to relational databases was “promoted”
 - used to be in the *.extlibx package, now in *.extlib
 - no longer “experimental”
- Nice video introduction by Brian Gleeson:
 - [https://www.openntf.org/main.nsf/blog.xsp?
permaLink=NHEF-9N7CKD](https://www.openntf.org/main.nsf/blog.xsp?permaLink=NHEF-9N7CKD)



Setting Up Your Database

- Use the JDBC Driver Plugin Wizard (DDE Tools menu) to bundle your JDBC driver as a plugin to install on Domino
 - TIP: you might need to include a license file with your driver
 - install via Update Site like you did with ExtLib and SocialSDK



Setting Up Your Database



- In **XSP Properties** of your database (Page Generation tab), make sure com.ibm.xsp.extlib.relational.library is checked
- In Package Explorer, create a folder called "**jdbc**" under **WebContent/WEB-INF** in your database
- Then create a new file in that folder called MyDbConnection.jdbc
 - or whatever name you want, with a .jdbc extension

Setting Up Your Database



The screenshot shows a Java-based IDE interface. On the left is a project navigation pane with the following structure:

- CustomControls
- Data
- Folders
- Forms
- Framesets
- Pages
- Resources
- SharedElements
- Views
- WebContent
 - WEB-INF
 - jdbc
 - MyDbConnection.jdbc
 - lib
 - faces-config.xml
 - xsp.properties

The file `MyDbConnection.jdbc` is selected in the navigation pane, and its contents are displayed in the main editor area:

```
<jdbc>
    <driver>org.apache.derby.jdbc.EmbeddedDriver</driver>
    <url>jdbc:derby:c:/temp/DominoTest.db</url>
    <user></user>
    <password></password>
</jdbc>
```

Make Every Moment Count



Relational Database XPage



- Now you can create an XPage with a JDBCQuery data source:
 - connectionName: MyDbConnection
 - sqlTable: MOVIEINFO
 - lots of other settings you can make too (query, etc.)
 - hover over the setting in the Preferences tab for inline help
- We can display the data in the same kind of dataTable we used earlier

Make Every Moment Count

Relational Database XPage



```
<xp:this.data>
    <xe:jdbcQuery var="jdbcQuery1" connectionName="MyDbConnection" sqlTable="MOVIEINFO" />
</xp:this.data>

<xp:dataTable rows="30" id="movieTable" value="#{jdbcQuery1}" var="movie" style="width:50%">
    <xp:column id="titleColumn">
        <xp:text escape="true" id="titleField" value="#{movie.title}">
            </xp:text>
    </xp:column>
    <xp:column id="studioColumn">
        <xp:text escape="true" id="studioField" value="#{movie.studio}">
            </xp:text>
    </xp:column>
    <xp:column id="yearColumn">
        <xp:text escape="true" id="yearField" value="#{movie.releaseyear}">
            </xp:text>
    </xp:column>
</xp:dataTable>
```

Relational Database XPage



Screenshot of a web browser showing a relational database table titled "MovieData.r". The table lists movie titles, studios, and release years.

Movie Title	Studio	Release Year
Gone With The Breeze	NGN	1939
Star Battles	Foz	1977
The Sound Of MP3s	Foz	1965
I.T.: The Computer Guys	Univixen	1982
Titantuan	Parallax	1997
The Ten Command Lines	Parallax	1956
Jowls	Univixen	1975
Doctor Zorro	NGN	1965
The Exerciser	Warmer Toes	1973
Snow Crash And The Seven Chores	Dizzy	1937
101 Damnations	Dizzy	1961
The Umpire Strikes Out	Foz	1980
Ben-Herd	NGN	1959
AvAtari	Foz	2009
Return Of The Gerbil	Foz	1983
Geriatric Park	Univixen	1993
Star Battle: Episode 1 - The Dennis Menace	Foz	1999
The Lying King	Dizzy	1994
The Stinger	Univixen	1973
Raiders Of The Beer Fridge	Parallax	1981

Make Every Moment Count

Paul Calhoun Has More!!!



- For a lot more detail (and sage advice), go to Paul Calhoun's session later this week

Make Every Moment Count

BP-1617: Relational XPages!!
Using XPages as the Presentation
Layer for RDBMS Data



Connect2016

The Premier Social Business and Digital Experience Conference



Make Every Moment Count



The Premier Social Business and Digital Experience Conference



Learning Resources

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count

Learning Java



- The book “Head First Java” is still excellent
- TLCC has some Notes-specific course material
- The only way to learn it is to use it
- Keep in mind that Notes/Domino is still (for now) on Java 6
 - cool new Java 7 and Java 8 features aren’t yet available to us

Make Every Moment Count



Learning XPages

- The book “Mastering XPages” sold by IBM Press
 - now in its second edition
- XPages Portable Command Guide
- <http://xpages.info>
- <http://www.notesin9.com/>
- If you get stuck, ask questions on <http://stackoverflow.com>
 - but please search for an answer first

Make Every Moment Count



Thank you

Connect 2016

The Premier Social Business and Digital Experience Conference

#ibmconnect

Make
Every
Moment
Count



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.



Acknowledgements and Disclaimers cont.

© Copyright IBM Corporation 2016. 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, Lotus, IBM Domino, IBM Notes, WebSphere, and DB2 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
- Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

Make Every Moment Count