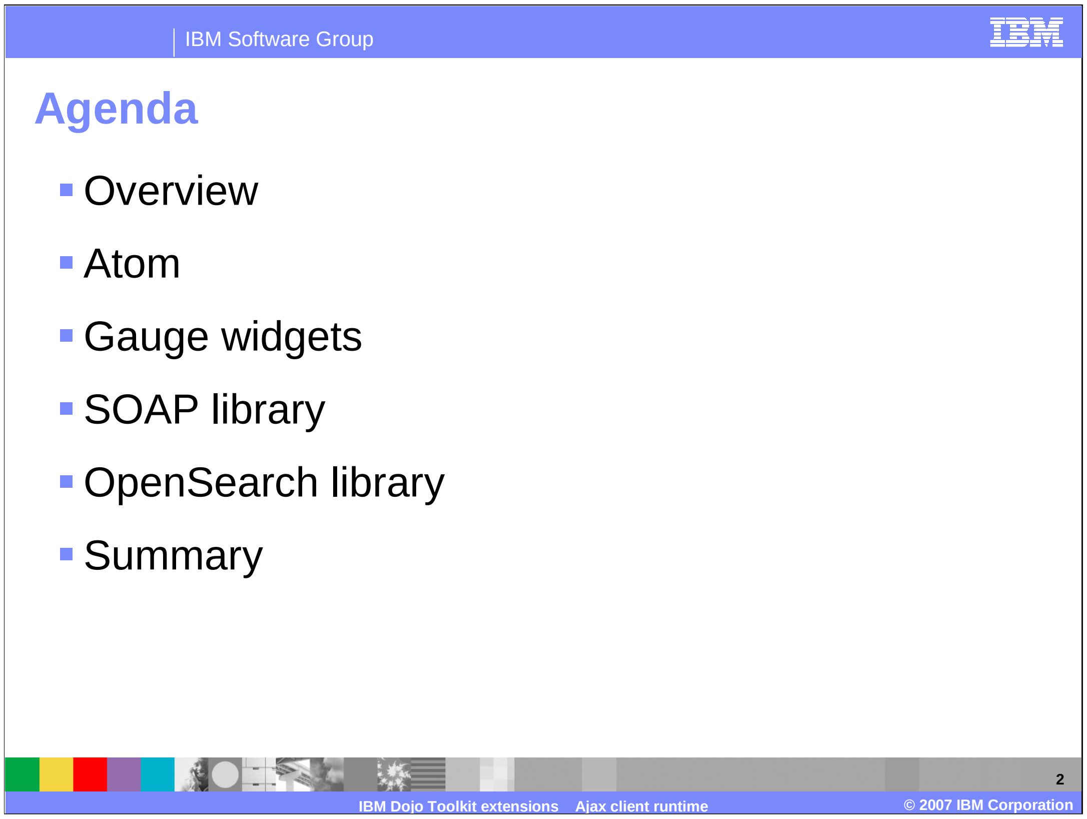
IBM Software Group

IBM® WebSphere® Application Server Feature Pack for Web 2.0

IBM Dojo toolkit extensions – Ajax client runtime

Updated September 29, 2008

This presentation will focus on IBM Dojo Toolkit Extensions also referred to as Ajax client runtime. It is a major piece of IBM WebSphere Application Server Feature Pack for Web 2.0



Agenda

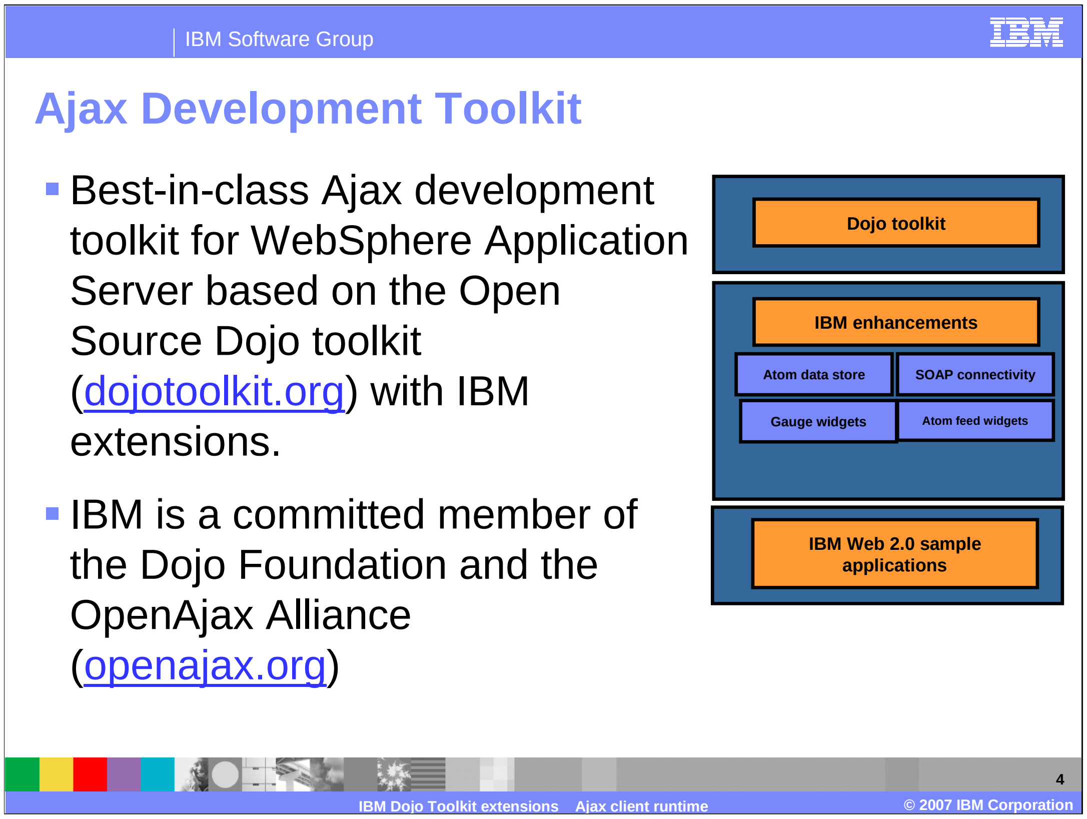
- Overview
- Atom
- Gauge widgets
- SOAP library
- OpenSearch library
- Summary

The presentation will cover the overview of the IBM Dojo Toolkit Extensions and the key pieces of the extensions. The IBM Dojo Toolkit Extensions include Atom Data access, Gauge widget, SOAP connectivity, and the OpenSearch library. This presentation will also summarize the importance of the extensions.

Section

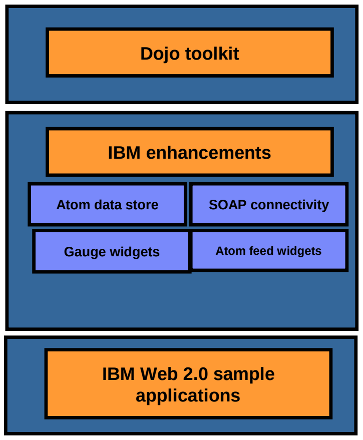
Overview

This next section will cover the overview of IBM Dojo Toolkit Extensions


IBM Software Group

Ajax Development Toolkit

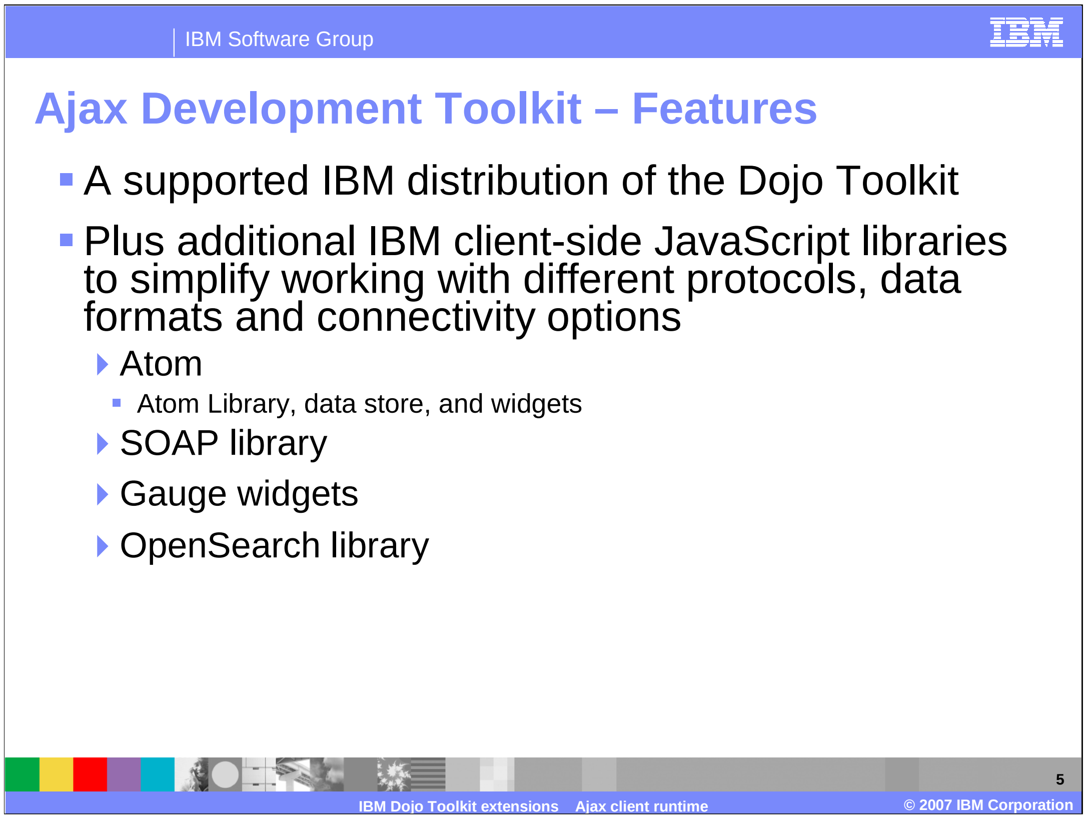
- Best-in-class Ajax development toolkit for WebSphere Application Server based on the Open Source Dojo toolkit (dojotoolkit.org) with IBM extensions.
- IBM is a committed member of the Dojo Foundation and the OpenAjax Alliance (openajax.org)



4

IBM Dojo Toolkit extensions   Ajax client runtime   © 2007 IBM Corporation

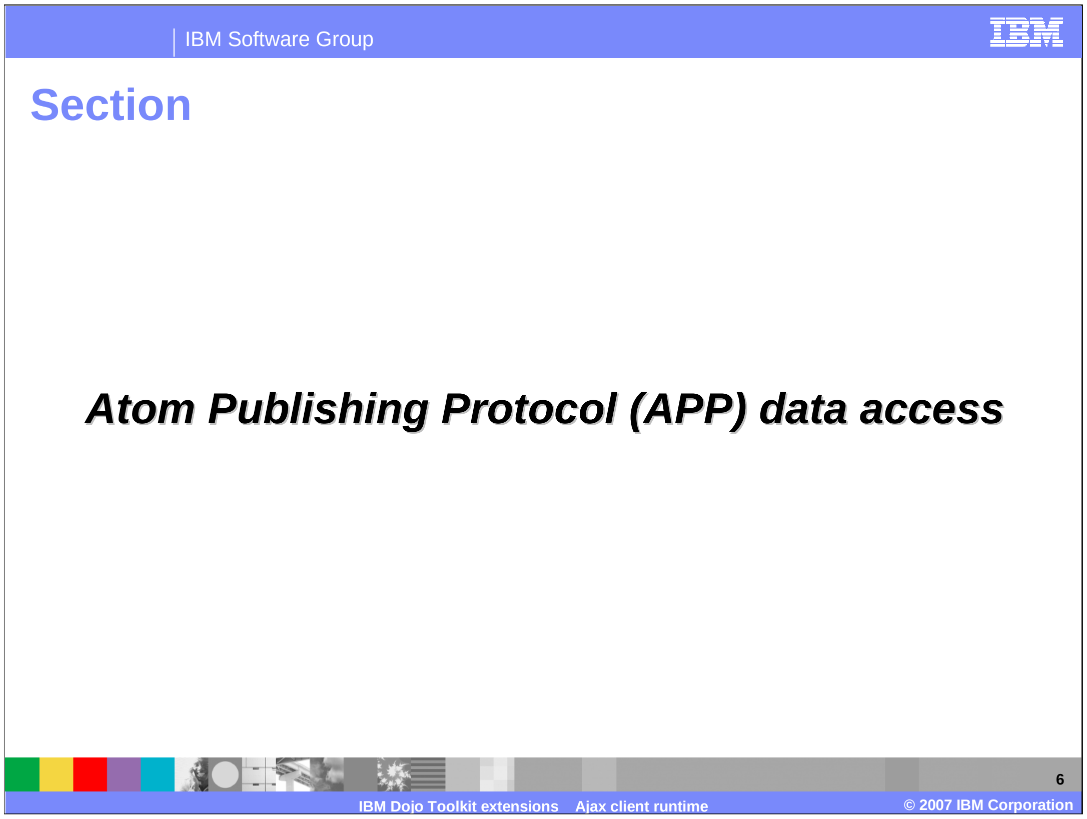
Ajax Development Toolkit is the best-in-class Ajax development toolkit for WebSphere Application Server based on the Open Source Dojo toolkit (dojotoolkit.org). IBM is a committed member of the Dojo Foundation and the OpenAjax Alliance (openajax.org).



Ajax Development Toolkit – Features

- A supported IBM distribution of the Dojo Toolkit
- Plus additional IBM client-side JavaScript libraries to simplify working with different protocols, data formats and connectivity options
  - Atom
    - Atom Library, data store, and widgets
  - SOAP library
  - Gauge widgets
  - OpenSearch library

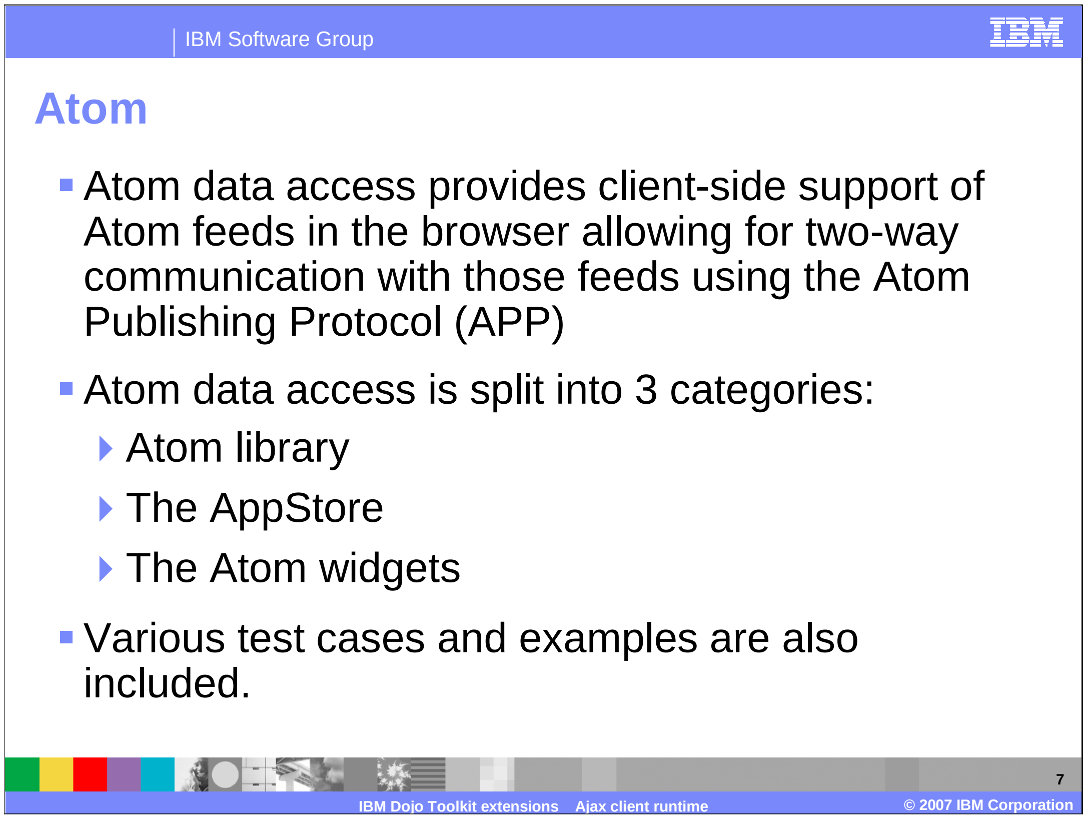
The features of the Ajax Development Toolkit include Ajax client runtime and IBM sample applications. Ajax client runtime is a distribution of the Dojo Toolkit version 1.0 with additional IBM enhancements to simplify working with different protocols, data formats and connectivity options. Therefore, Ajax client runtime consists of Dojo toolkit 1.0 and IBM Dojo toolkit Extensions which are Atom, SOAP library, Gauge widgets and OpenSearch library.



Section

Atom Publishing Protocol (APP) data access

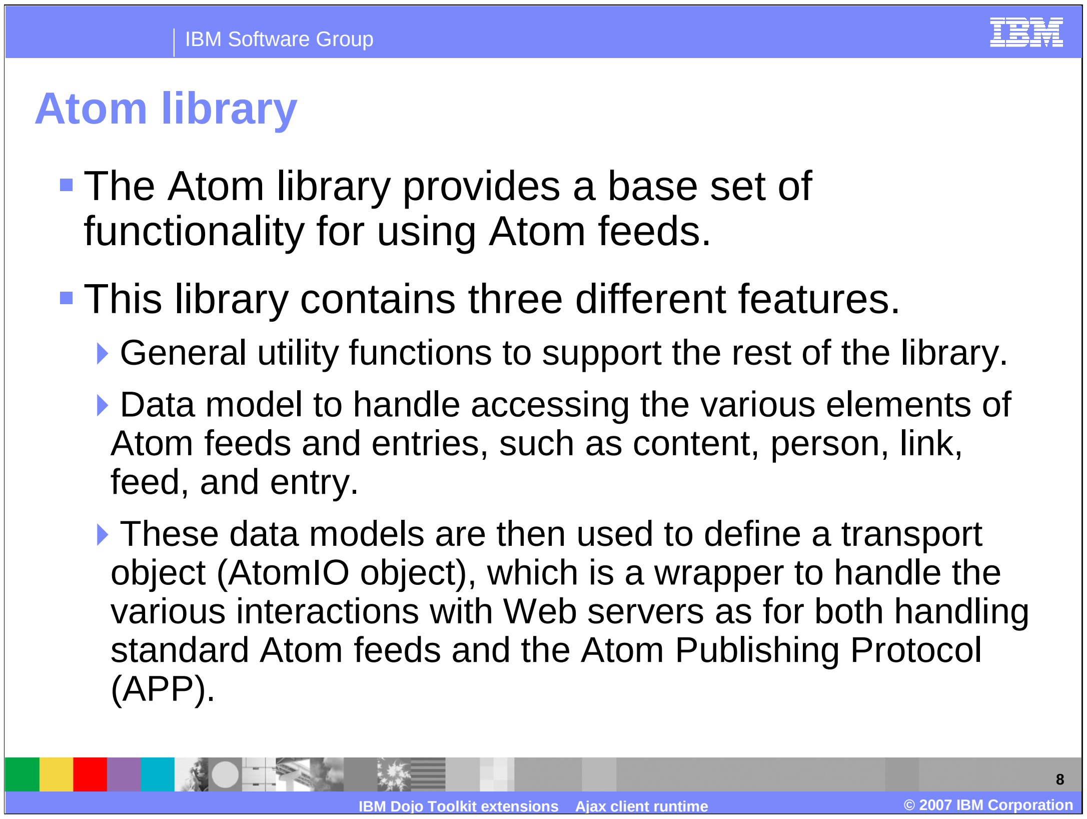
This next section will talk about Atom Publishing Protocol (APP) data access.



Atom

- Atom data access provides client-side support of Atom feeds in the browser allowing for two-way communication with those feeds using the Atom Publishing Protocol (APP)
- Atom data access is split into 3 categories:
  - Atom library
  - The AppStore
  - The Atom widgets
- Various test cases and examples are also included.

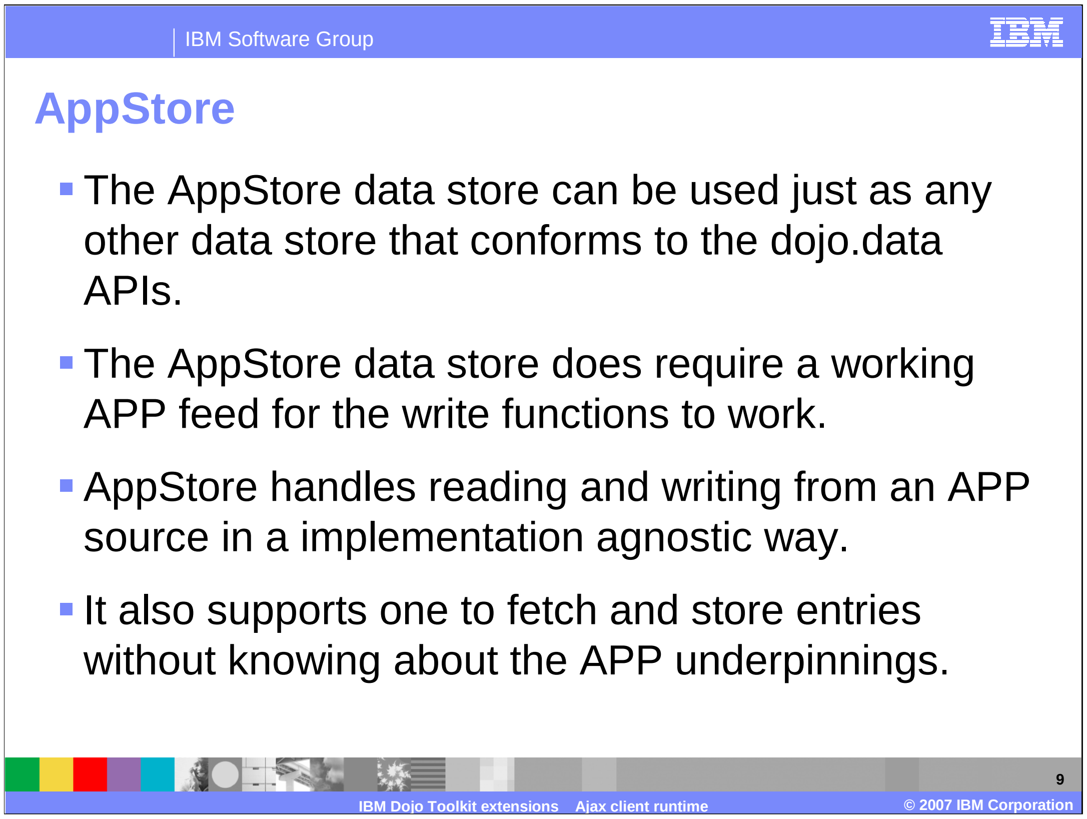
The Atom Data Access provides client-side support of Atom feeds in the browser allowing for a two-way communication with those feeds using the Atom Publishing Protocol (APP). Atom Data Access has three principal parts: the Atom library, the AppStore, and the Atom widgets. Various test cases and examples are also included.



Atom library

- The Atom library provides a base set of functionality for using Atom feeds.
- This library contains three different features.
  - General utility functions to support the rest of the library.
  - Data model to handle accessing the various elements of Atom feeds and entries, such as content, person, link, feed, and entry.
  - These data models are then used to define a transport object (AtomIO object), which is a wrapper to handle the various interactions with Web servers as for both handling standard Atom feeds and the Atom Publishing Protocol (APP).

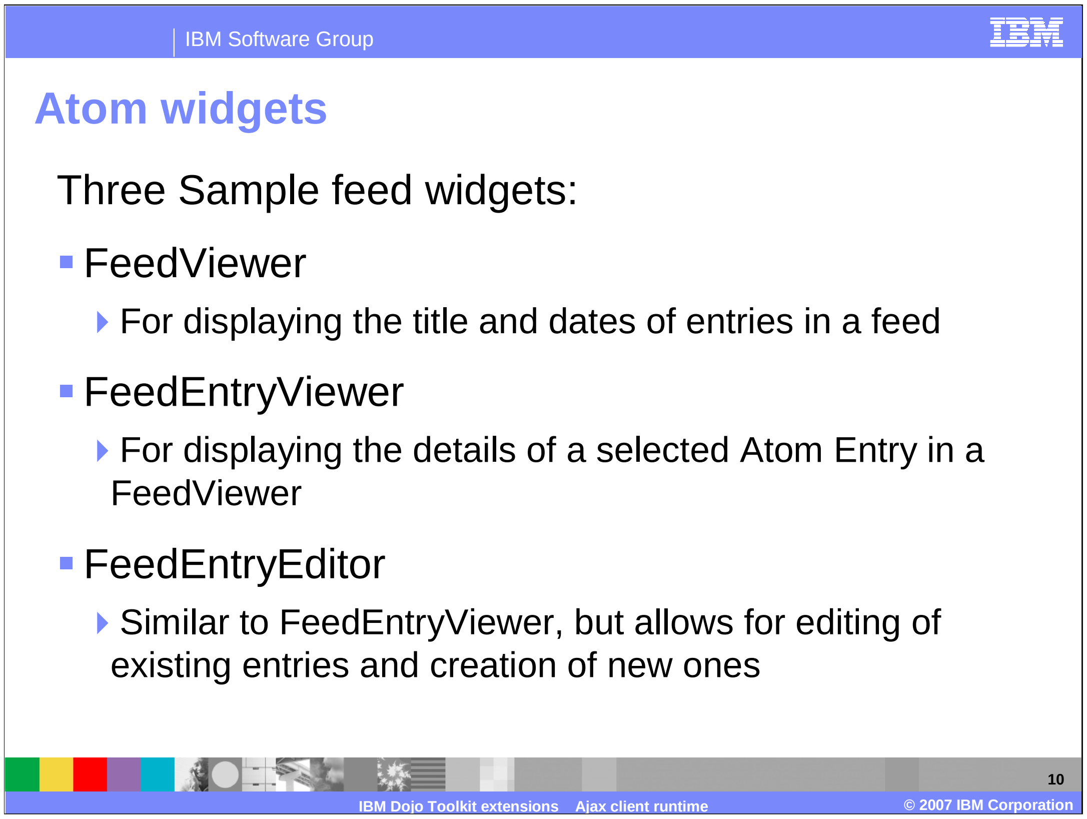
The Atom library: This library contains three different features. First are the general utility functions to support the rest of the library. Next is the data model for the various parts of Atom, such as content, person, link, feed, and entry. These data models are then used to define the AtomIO object, which is a wrapper to the various functions intrinsic to Atom feeds and the Atom Publishing Protocol (APP).



AppStore

- The AppStore data store can be used just as any other data store that conforms to the dojo.data APIs.
- The AppStore data store does require a working APP feed for the write functions to work.
- AppStore handles reading and writing from an APP source in a implementation agnostic way.
- It also supports one to fetch and store entries without knowing about the APP underpinnings.

The AppStore: Implementing the dojo.data.api API Read, Identity, and Write API's, the AppStore handles reading and writing from an APP source in an implementation agnostic way. It also supports one to fetch and store entries without knowing about the APP underpinnings.



IBM Software Group

Atom widgets

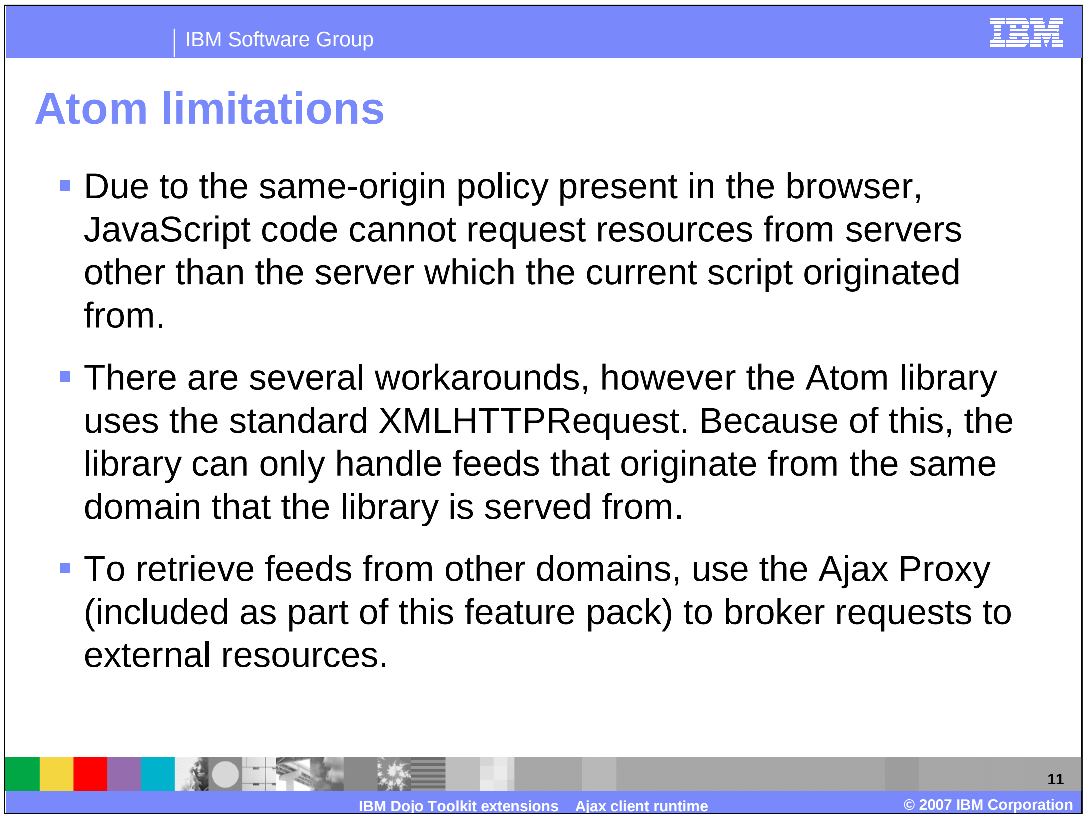
Three Sample feed widgets:

- FeedViewer
  - For displaying the title and dates of entries in a feed
- FeedEntryViewer
  - For displaying the details of a selected Atom Entry in a FeedViewer
- FeedEntryEditor
  - Similar to FeedEntryViewer, but allows for editing of existing entries and creation of new ones

10

IBM Dojo Toolkit extensions   Ajax client runtime   © 2007 IBM Corporation

As mentioned earlier, Atom widgets are sample feed widgets which use the Atom (APP) Data Access feature. These 3 widgets each perform a different function. **Feed Viewer Widget** is a reference widget for an Atom feed viewer implementation. This widget fetches a feed from the given URL and displays, in chronological order, entry titles and updated elements. This widget supports click events to highlight an entry and publishes the selected entry on the given topic. **Feed Entry Viewer Widget** displays an individual entry element. A bar is provided at the top with a menu to turn specific elements on or off. The viewable elements can be dictated at creation time. In addition to the functions outlined, there are several additional functions, used mainly internally, to set the value of various nodes in the page. **Feed Entry Editor Widget** displays the elements of an individual. It also allows edits to be made on this entry, if the entry has a link with a real attribute that equals to "edit". This widget does none of the work of sending the edits back as requests. Instead, the widget updates its rendering and sends the updates on the entrySelectionTopic and allows the FeedViewer widget to handle the sending of the data. As mentioned, this widget extends the FeedEntryEditor widget; therefore all of its variables and functions are inherited. However, this widget does override the previously mentioned undocumented functions that set the display node values, as it must replace the value with an editor object in the event that this entry is being edited. This widget has no additional variables or functions over the FeedEntryViewer.



Atom limitations

- Due to the same-origin policy present in the browser, JavaScript code cannot request resources from servers other than the server which the current script originated from.
- There are several workarounds, however the Atom library uses the standard XMLHTTPRequest. Because of this, the library can only handle feeds that originate from the same domain that the library is served from.
- To retrieve feeds from other domains, use the Ajax Proxy (included as part of this feature pack) to broker requests to external resources.

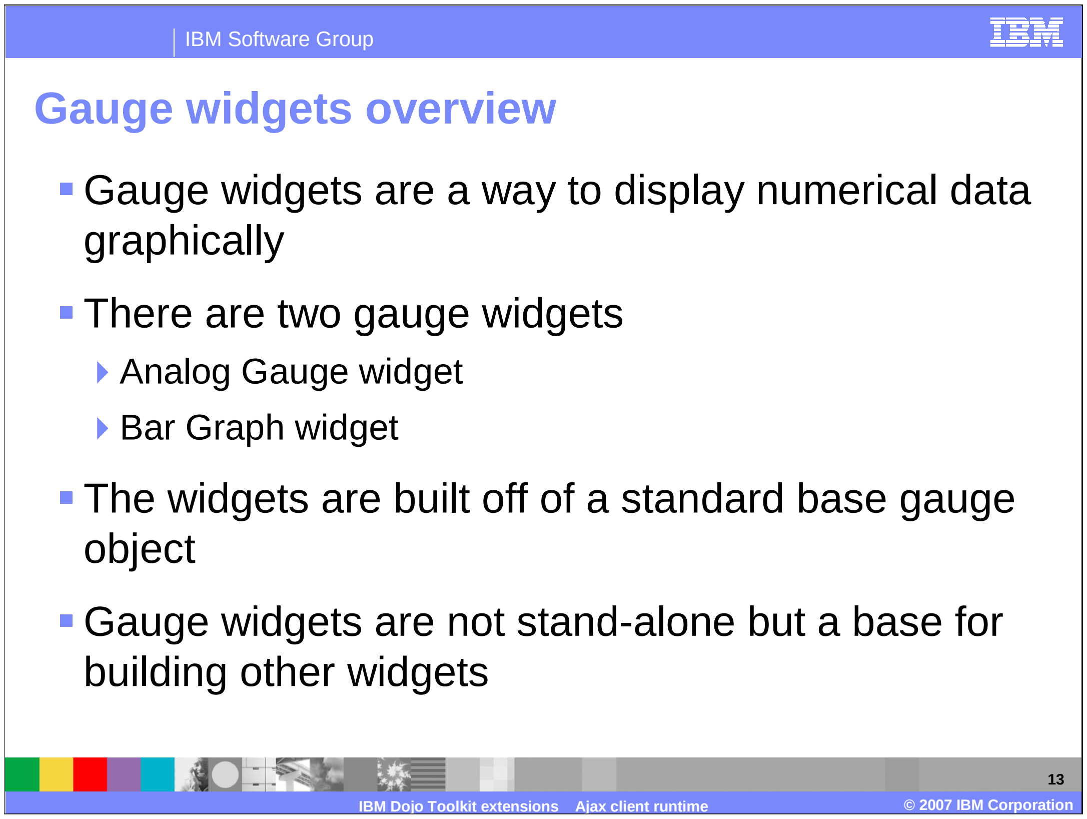
Due to the same-origin policy present in the browser, JavaScript code cannot request resources from servers other than the server which the current script originated from.

There are several workarounds, however the Atom library uses the standard XMLHTTPRequest. Because of this, the library can only handle feeds that originate from the same domain that the library is served from. To retrieve feeds from other domains, use the Ajax Proxy (included as part of this feature pack) to broker requests to external resources.

Section

Gauge widgets

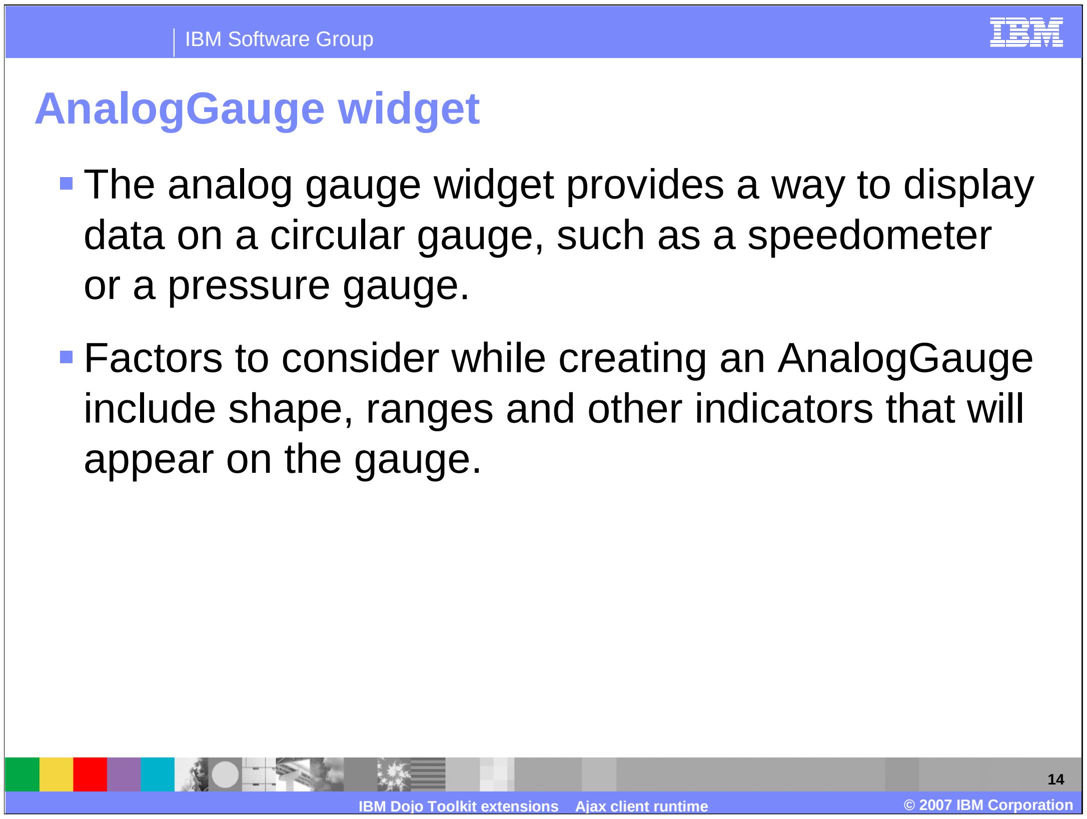
The next section will cover the Gauge widgets.



Gauge widgets overview

- Gauge widgets are a way to display numerical data graphically
- There are two gauge widgets
  - Analog Gauge widget
  - Bar Graph widget
- The widgets are built off of a standard base gauge object
- Gauge widgets are not stand-alone but a base for building other widgets

The gauge library includes a pair of widgets for displaying numerical data in a graphically rich way. Using Scalable Vector Graphics (SVG) or Vector Markup Language (VML) depending on the browser, the AnalogGauge and BarGraph widgets display numerical data with customizable ranges, tick marks, and indicators at any size. The ability to connect these gauges to an updating data source (such as the Web Messaging Service) allows for creating self-updating graphical displays. The widgets are built off of a standard base gauge object and they are not stand-alone but a base for building other widgets

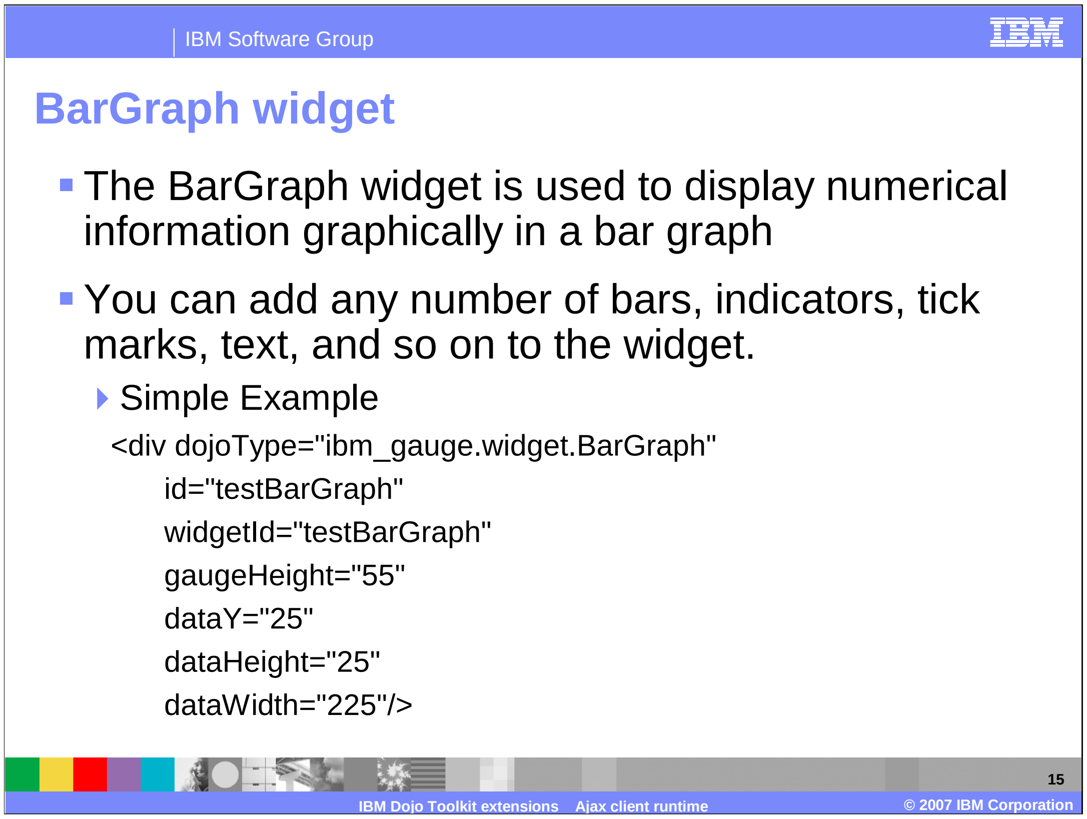


AnalogGauge widget

- The analog gauge widget provides a way to display data on a circular gauge, such as a speedometer or a pressure gauge.
- Factors to consider while creating an AnalogGauge include shape, ranges and other indicators that will appear on the gauge.

The AnalogGauge widget provides a way to display data on a circular gauge, such as a speedometer or a pressure gauge. To create an AnalogGauge, one must first determine the shape they want the gauge to take, what indicators will appear on the gauge, what ranges will exist on the gauge, and other factors.

The above example code creates a new analog gauge widget, housed within the div with an ID of 'testGauge'. The gauge currently has no ranges, indicators, or tick marks. However, the data area for the gauge is indicated by the parameters that were passed in.

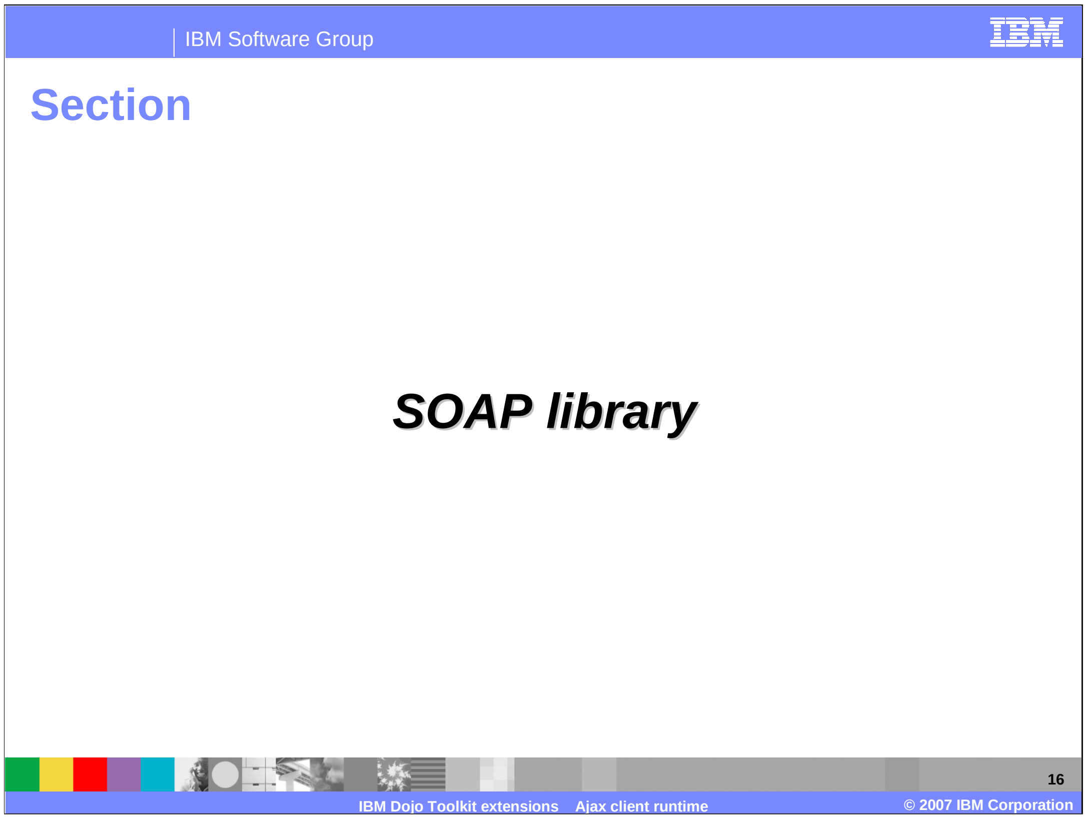


BarGraph widget

- The BarGraph widget is used to display numerical information graphically in a bar graph
- You can add any number of bars, indicators, tick marks, text, and so on to the widget.
  - Simple Example

```
<div dojoType="ibm_gauge.widget.BarGraph"
     id="testBarGraph"
     widgetId="testBarGraph"
     gaugeHeight="55"
     dataY="25"
     dataHeight="25"
     dataWidth="225"/>
```

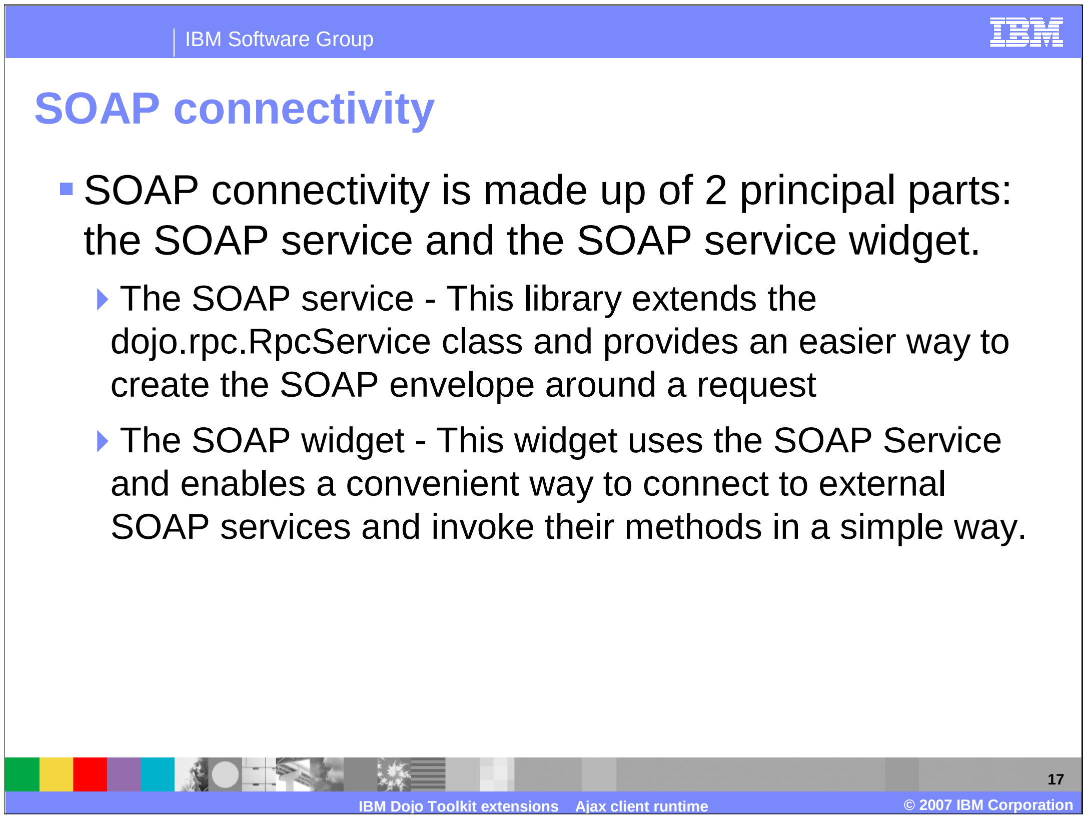
The BarGraph widget is used to display numerical information graphically in a bar graph. The possibilities here are just as endless as with the AnalogGauge widget. Any number of bars, indicators, tick marks, text and so on, can be added to the widget as shown in the simple example given.



Section

SOAP library

The next section will discuss SOAP connectivity


IBM Software Group

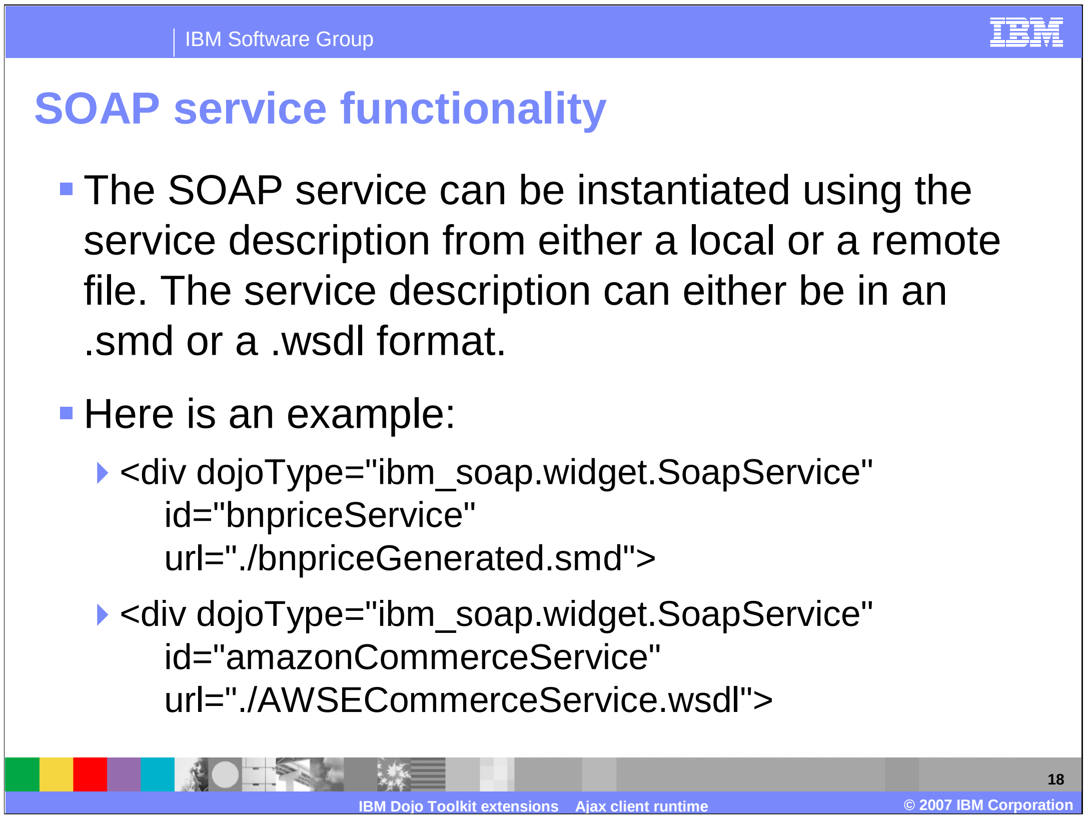
SOAP connectivity

- SOAP connectivity is made up of 2 principal parts: the SOAP service and the SOAP service widget.
  - The SOAP service - This library extends the dojo.rpc.RpcService class and provides an easier way to create the SOAP envelope around a request
  - The SOAP widget - This widget uses the SOAP Service and enables a convenient way to connect to external SOAP services and invoke their methods in a simple way.

17

IBM Dojo Toolkit extensions   Ajax client runtime   © 2007 IBM Corporation

SOAP connectivity makes it easier to invoke public SOAP-based Web services from Ajax applications. SOAP connectivity is comprised of 2 principal parts: the SOAP service and the SOAP service widget. The SOAP Service library extends the dojo.rpc.RpcService class and provides an easier way to create the SOAP envelope around a request. The SOAP Widget uses the SOAP Service and enables a convenient way to connect to external SOAP services and invoke their methods in a simple way. The SOAP service can be instantiated using the service description from either a local or a remote file. The service description can be either in a Simple Method Description (.smd) format or a WSDL (.wsdl) format.


IBM Software Group

SOAP service functionality

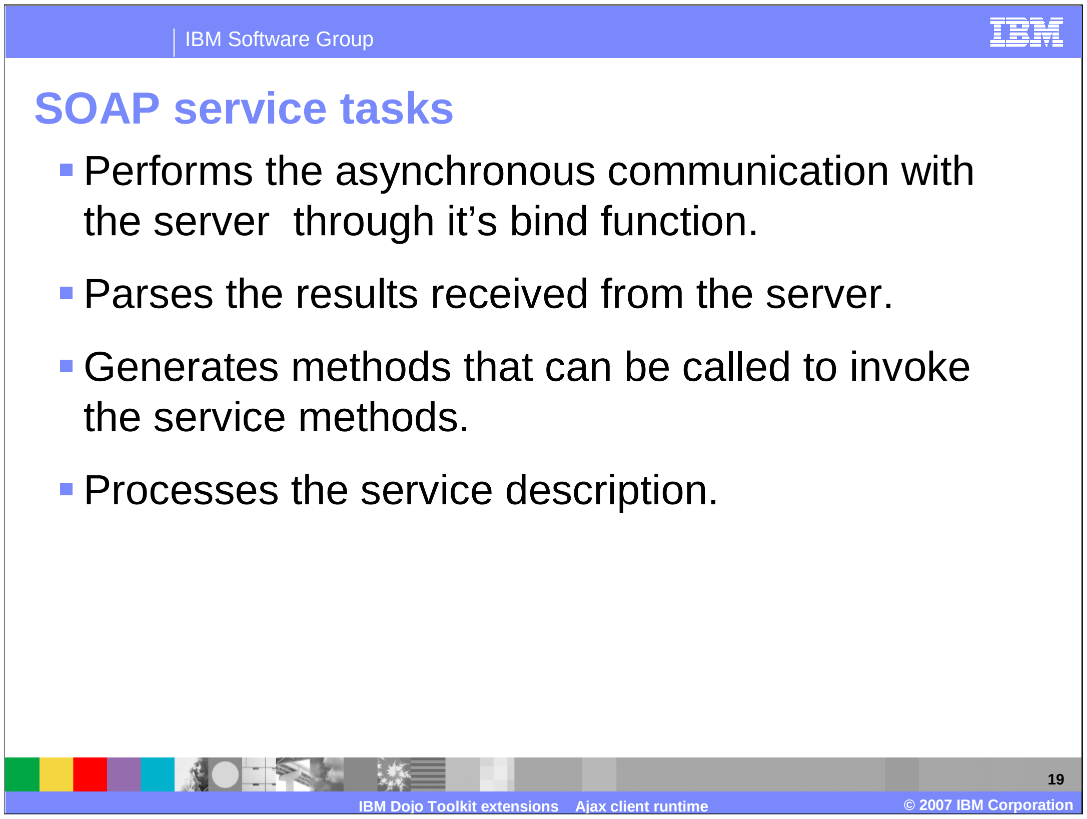
- The SOAP service can be instantiated using the service description from either a local or a remote file. The service description can either be in an .smd or a .wsdl format.
- Here is an example:
  - <div dojoType="ibm_soap.widget.SoapService"
        id="bnpriceService"
        url="./bnpriceGenerated.smd">
  - <div dojoType="ibm_soap.widget.SoapService"
        id="amazonCommerceService"
        url="./AWSECommerceService.wsdl">

18

IBM Dojo Toolkit extensions   Ajax client runtime   © 2007 IBM Corporation

The SOAP service can be instantiated using the service description from either a local or a remote file. The service description can either be in a .smd or a .wsdl format as shown in the example.

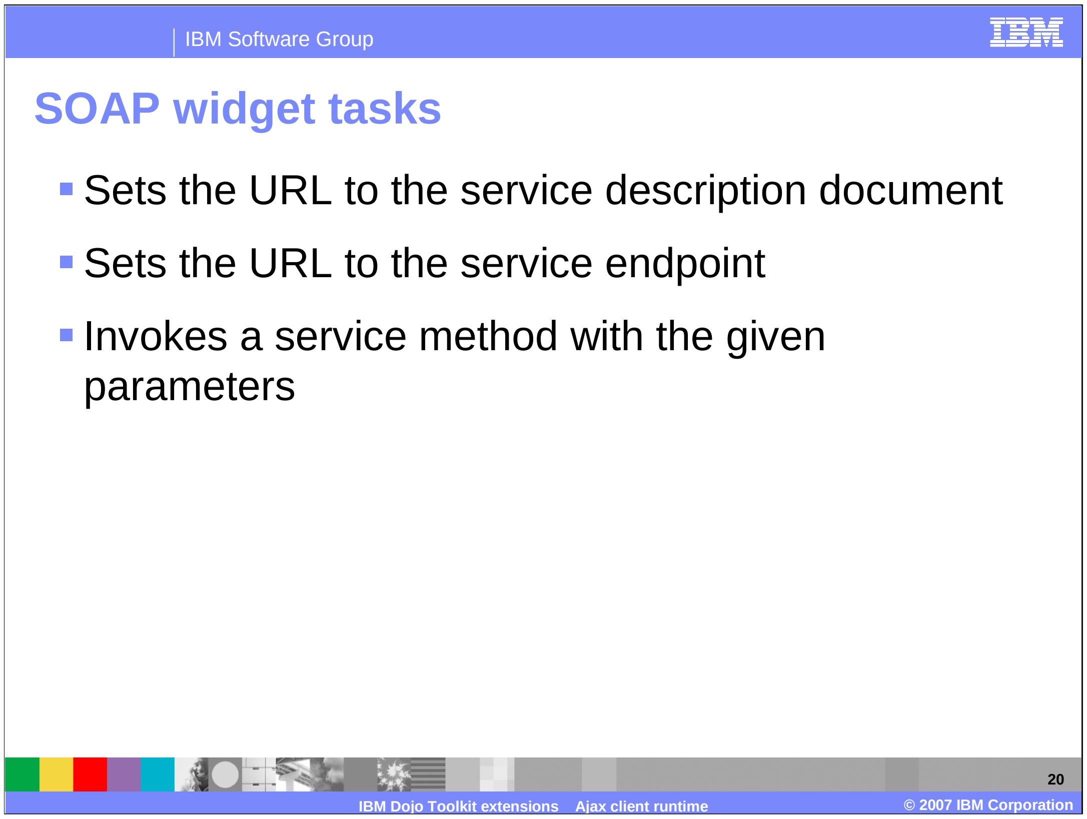
As you can see, you need to provide the location of the service description such as the URL. Once the widgets are parsed and instantiated, the service is ready for use. Typically, you will start with a service that provides a service description such as a .wsdl file. You can use the provided WsdlParser to convert it to .smd format. Note that a Simple Method Description file (.smd), is a simple JSON string that defines a URL. It will process the RPC requests, any methods available at that URL, and the parameters those methods take.



SOAP service tasks

- Performs the asynchronous communication with the server through it's bind function.
- Parses the results received from the server.
- Generates methods that can be called to invoke the service methods.
- Processes the service description.

SOAP Service performs several tasks. Some of its tasks includes performing the asynchronous communication with the server through it's bind function. It also parses the results received from the server and generates methods that can be called to invoke the service methods. Additionally, it processes the service description.


IBM Software Group

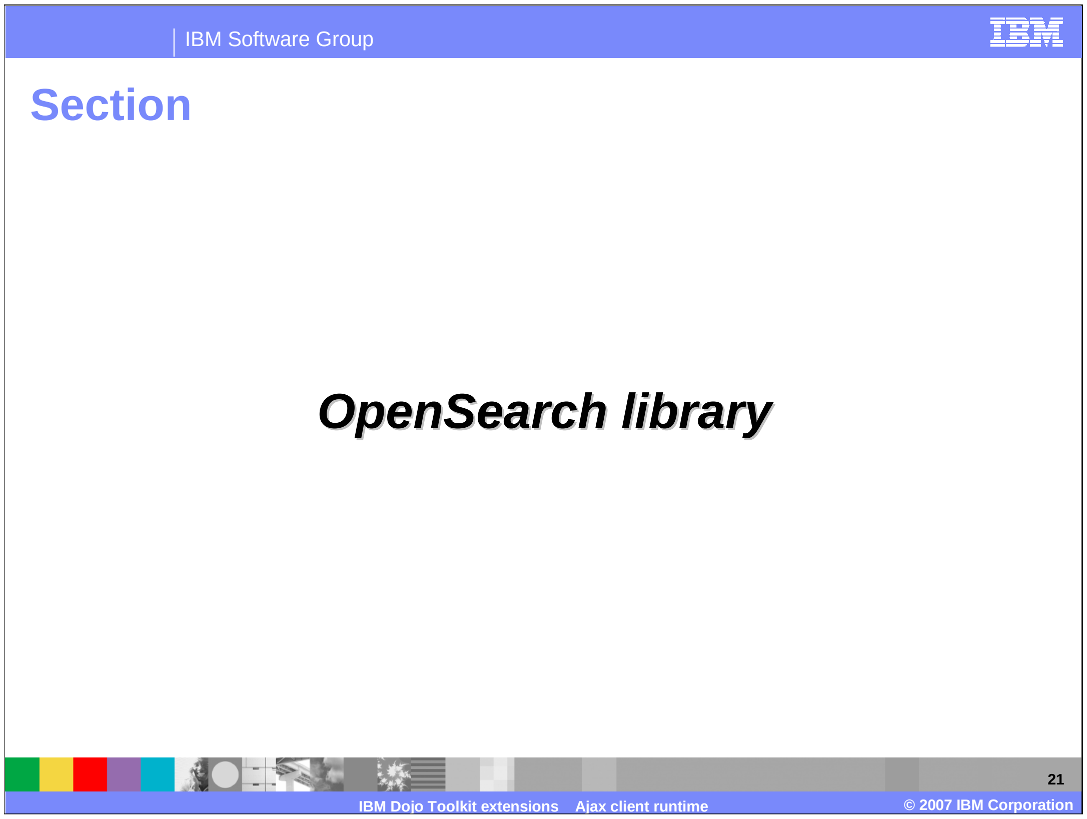
SOAP widget tasks

- Sets the URL to the service description document
- Sets the URL to the service endpoint
- Invokes a service method with the given parameters

20

IBM Dojo Toolkit extensions Ajax client runtime © 2007 IBM Corporation

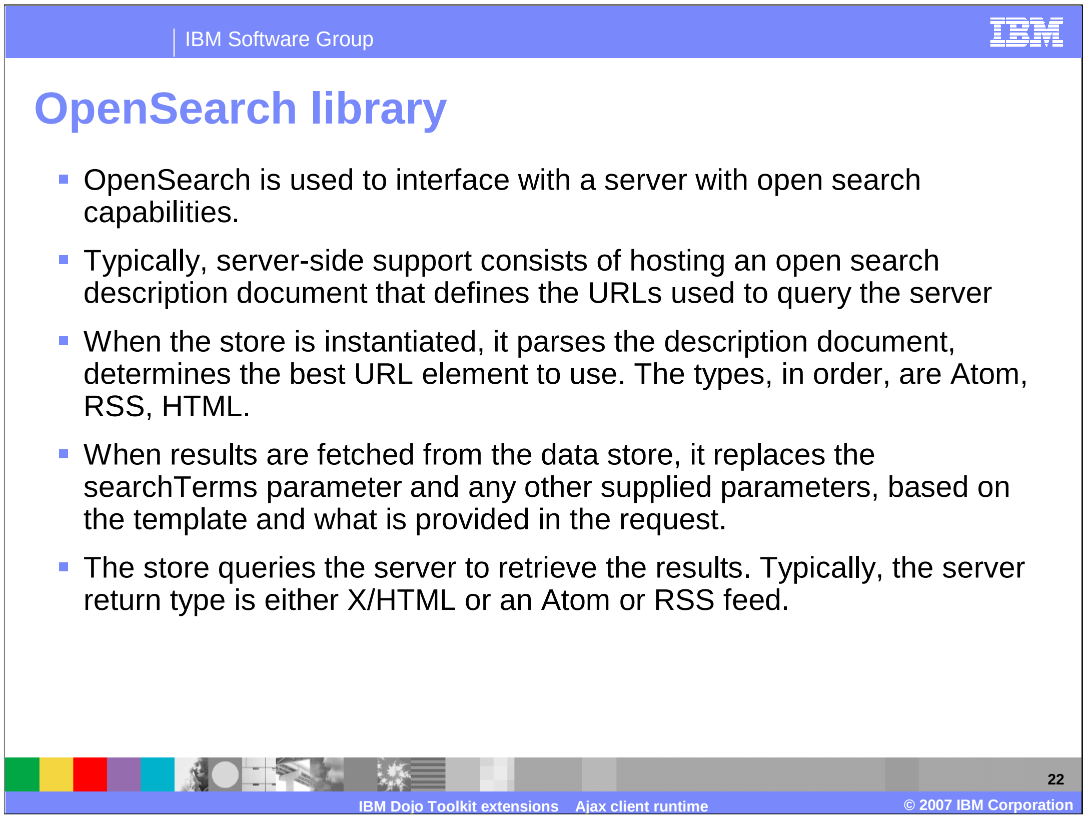
SOAP Service Widget tasks include setting the URL to the service description document and to the service endpoint. It also invokes a service method with the given parameters



Section

OpenSearch library

The next section will discuss OpenSearch Library



OpenSearch library

- OpenSearch is used to interface with a server with open search capabilities.
- Typically, server-side support consists of hosting an open search description document that defines the URLs used to query the server
- When the store is instantiated, it parses the description document, determines the best URL element to use. The types, in order, are Atom, RSS, HTML.
- When results are fetched from the data store, it replaces the searchTerms parameter and any other supplied parameters, based on the template and what is provided in the request.
- The store queries the server to retrieve the results. Typically, the server return type is either X/HTML or an Atom or RSS feed.

OpenSearch is used to interface with a server with open search capabilities.

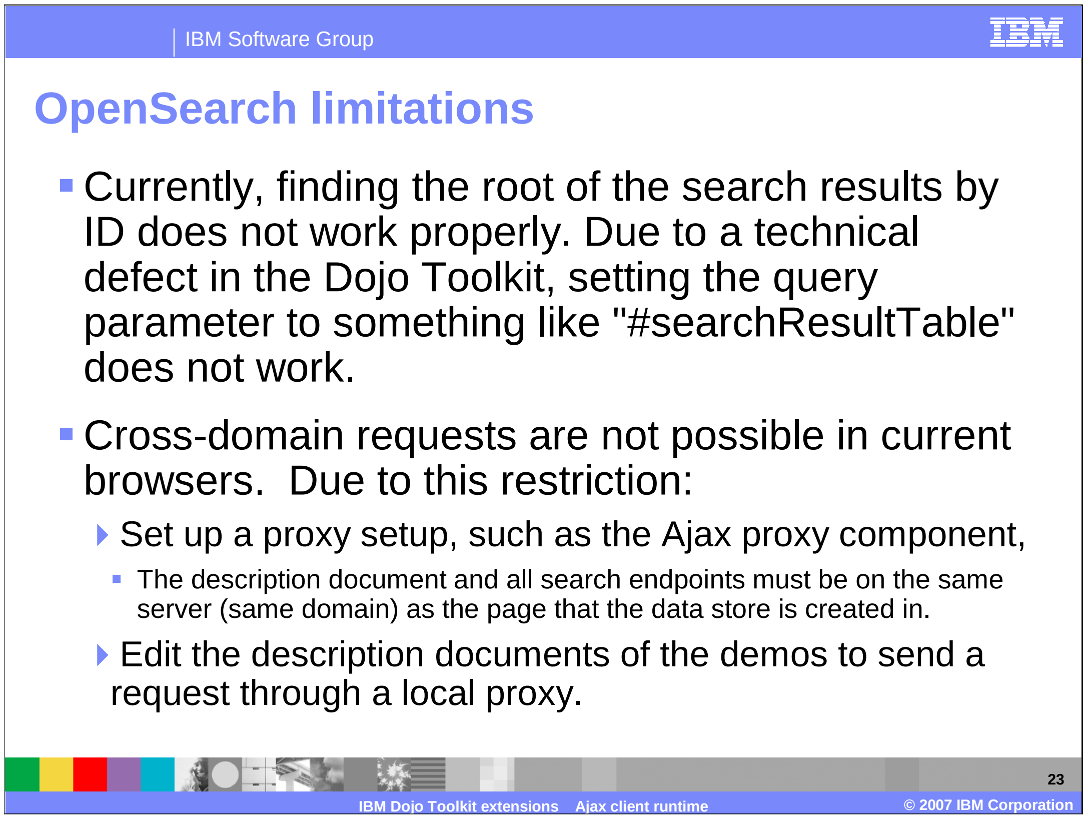
Typically, server-side support consists of hosting an open search description document that defines the URLs used to query the server.

When the store is instantiated, it parses the description document, and determines the best URL element to use. The types in order, are Atom, RSS, and HTML.

When results are fetched from the data store, it replaces the searchTerms parameter and any other supplied parameters, based on the template and what is provided in the request.

The store queries the server to retrieve the results.

Typically, the server return type is either XHTML or an Atom or RSS feed.


IBM Software Group

OpenSearch limitations

- Currently, finding the root of the search results by ID does not work properly. Due to a technical defect in the Dojo Toolkit, setting the query parameter to something like "#searchResultTable" does not work.
- Cross-domain requests are not possible in current browsers. Due to this restriction:
  - Set up a proxy setup, such as the Ajax proxy component,
    - The description document and all search endpoints must be on the same server (same domain) as the page that the data store is created in.
  - Edit the description documents of the demos to send a request through a local proxy.

23

IBM Dojo Toolkit extensions Ajax client runtime © 2007 IBM Corporation

Currently, finding the root of the search results by ID does not work properly. Due to a technical defect in the Dojo Toolkit, setting the query parameter to something like "#searchResultTable" does not work.

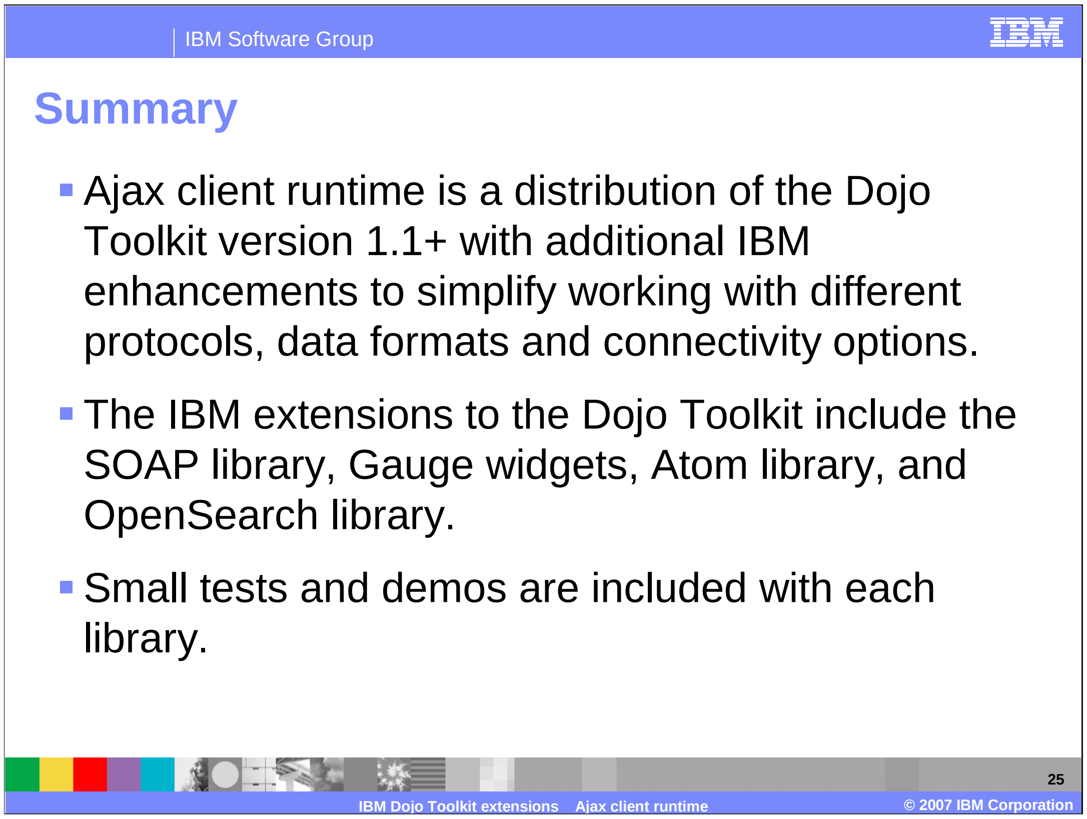
Cross-domain requests are not possible in current browsers. The data store is instantiated using an URL to a open search description document. Unless you have a proxy setup, such as the Ajax Proxy component included with this Feature Pack, this document and all search endpoints must be on the same server (same domain) as the page that the data store is created in, because of cross-domain limitations of browsers. The description document and all search endpoints must be on the same server (same domain) as the page that the data store is created in.

The demos reference OpenSearch description documents which must be fetched and edited to send the requests through a local proxy, as cross-domain requests are not possible in current browsers.

Section

Summary

The next section will summarize what you have learned so far on the IBM Dojo extensions.


IBM Software Group

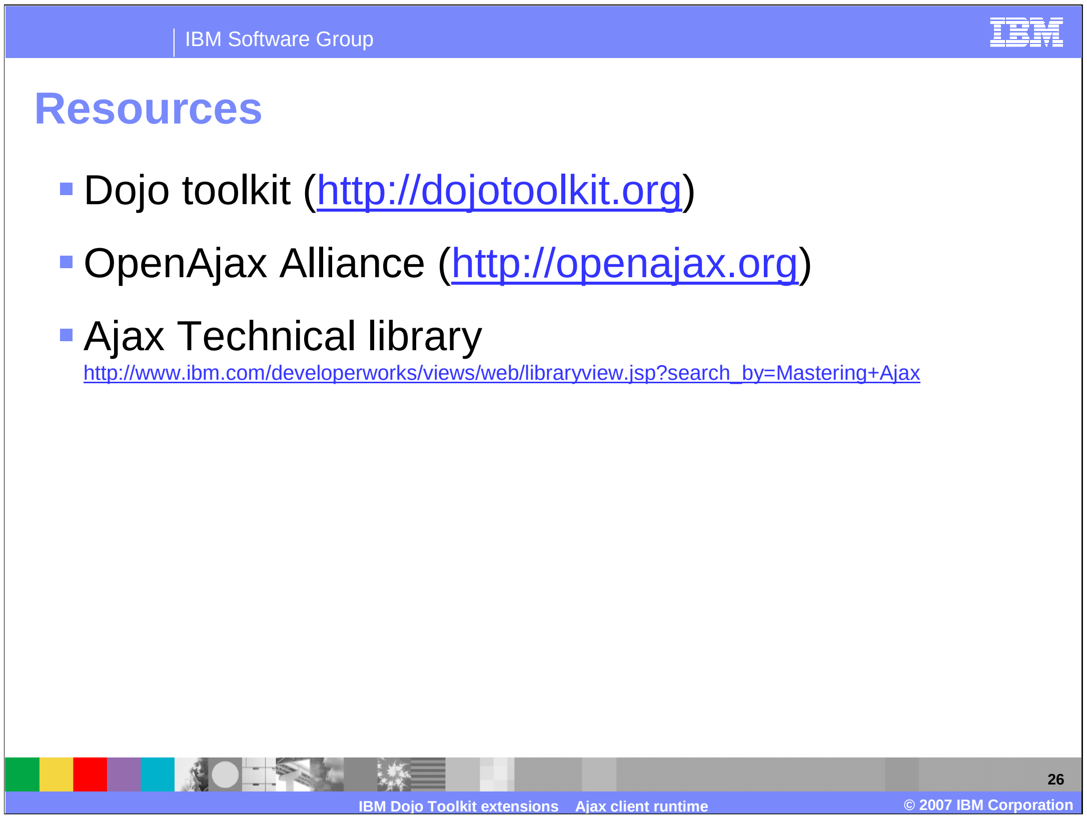
Summary

- Ajax client runtime is a distribution of the Dojo Toolkit version 1.1+ with additional IBM enhancements to simplify working with different protocols, data formats and connectivity options.
- The IBM extensions to the Dojo Toolkit include the SOAP library, Gauge widgets, Atom library, and OpenSearch library.
- Small tests and demos are included with each library.

25

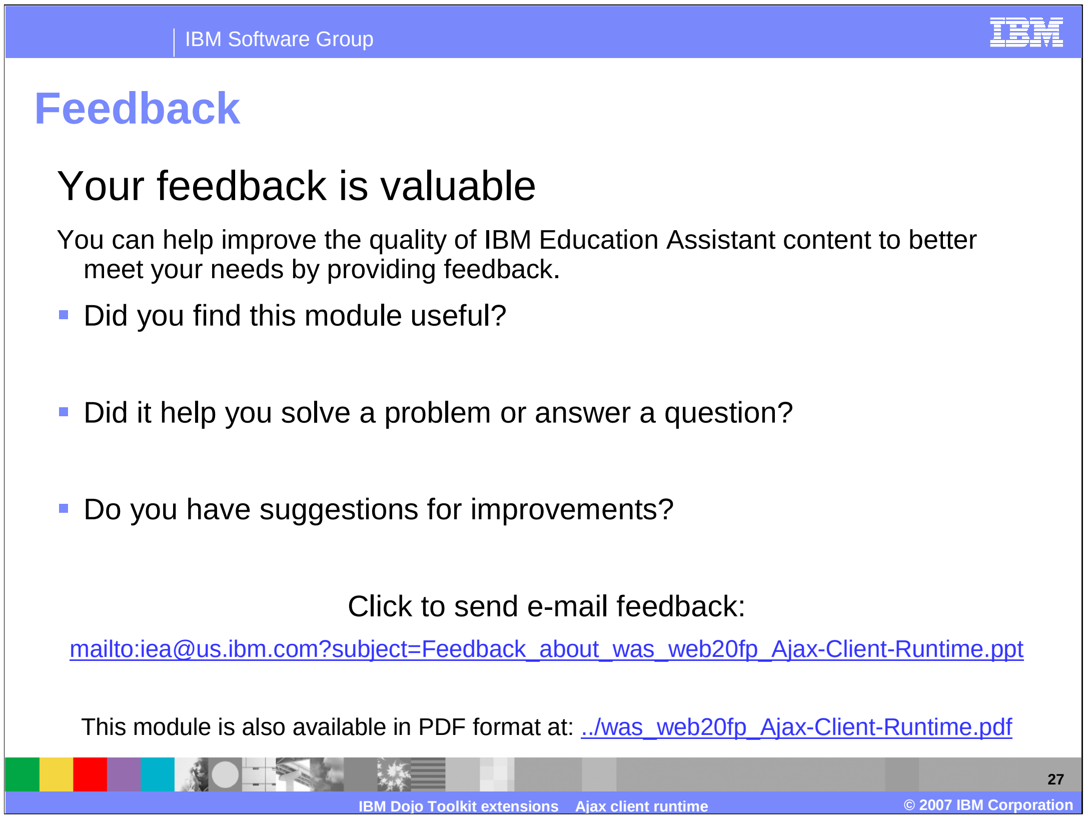
IBM Dojo Toolkit extensions   Ajax client runtime   © 2007 IBM Corporation

Ajax client runtime is a distribution of the Dojo Toolkit version 1.1+ with additional IBM enhancements to simplify working with different protocols, data formats and connectivity options. The main parts of the IBM Dojo Toolkit extensions include the Atom Data Access, the Gauge widgets and the SOAP library.



Resources

- Dojo toolkit (http://dojotoolkit.org)
- OpenAjax Alliance (http://openajax.org)
- Ajax Technical library
  http://www.ibm.com/developerworks/views/web/libraryview.jsp?search_by=Mastering+Ajax



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

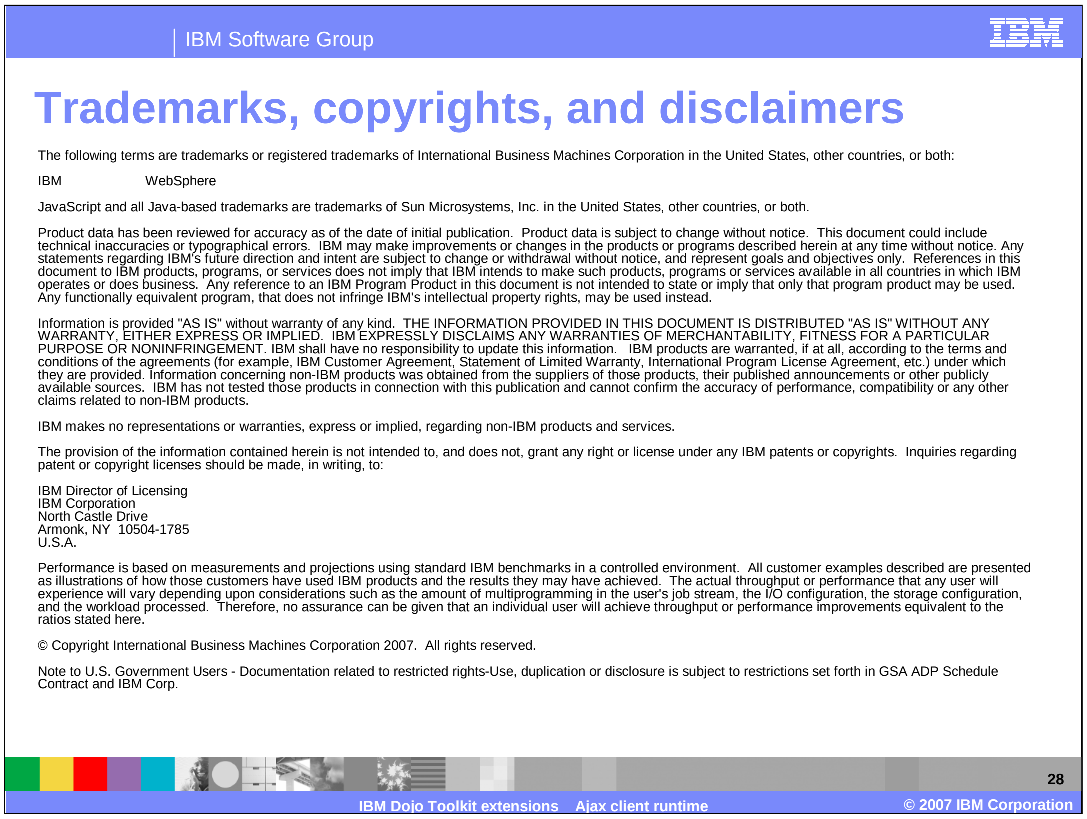
- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_was_web20fp_Ajax-Client-Runtime.ppt

This module is also available in PDF format at: ../was_web20fp_Ajax-Client-Runtime.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, copyrights, and disclaimers