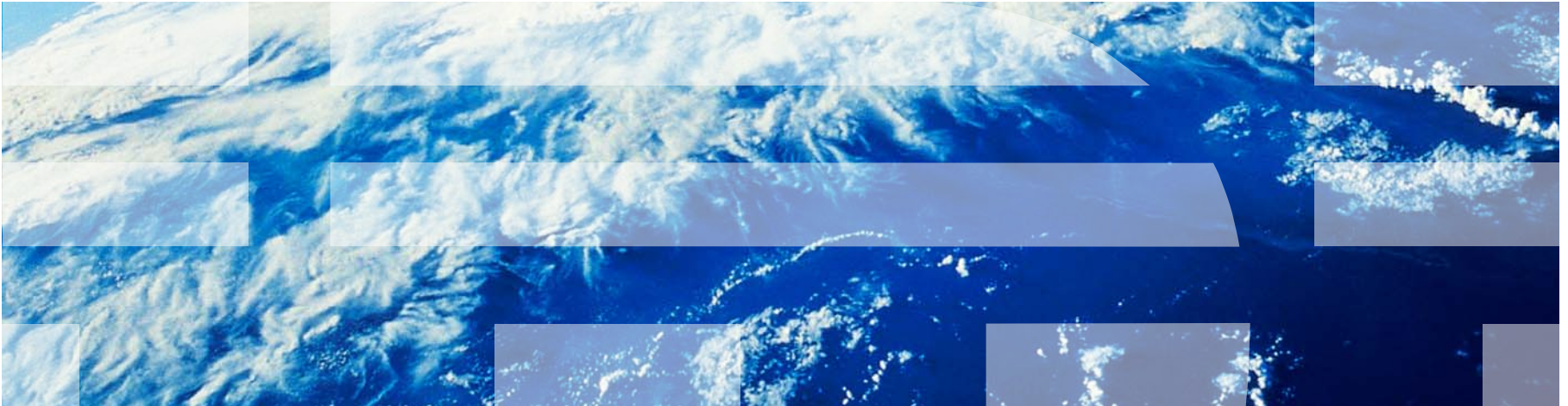


# ***IBM Worklight V6.1.0 Getting Started***

## **Creating an application with IBM Worklight Application Framework**



## ***Trademarks***

- IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Worklight is a trademark or registered trademark of Worklight, an IBM Company. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)” at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).
- Other company products or service names may be trademarks or service marks of others.
- This document may not be reproduced in whole or in part without the prior written permission of IBM.

## ***About IBM®***

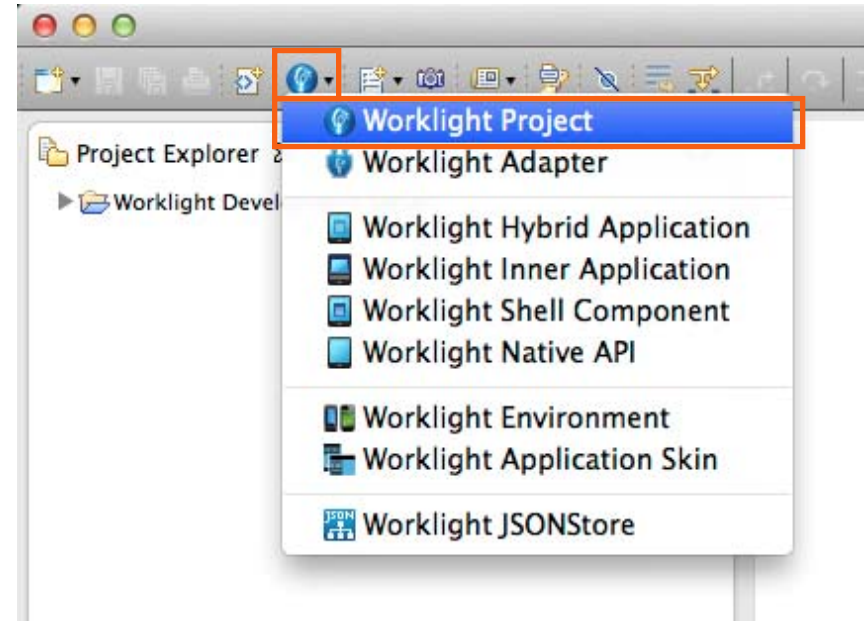
- See <http://www.ibm.com/ibm/us/en/>

# Agenda

- Creating a project with a hybrid application
- Weather application overview
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- Deploying and previewing

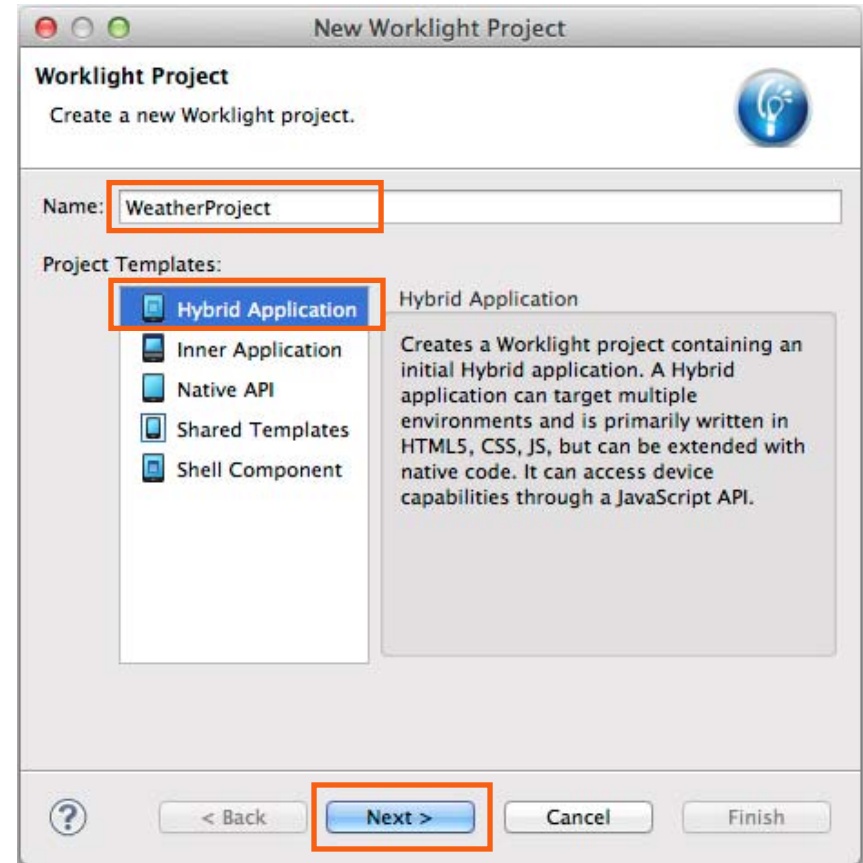
## New Worklight Project

- Create an IBM Worklight® project: in the toolbar, click **Create a Worklight Artifact > Worklight Project**
- Other ways to create an IBM Worklight project:
  - Click **File > New > Project > Worklight Project > Next**.
  - Right-click an empty space in the **Project Explorer**, and click **New > Worklight Project**.



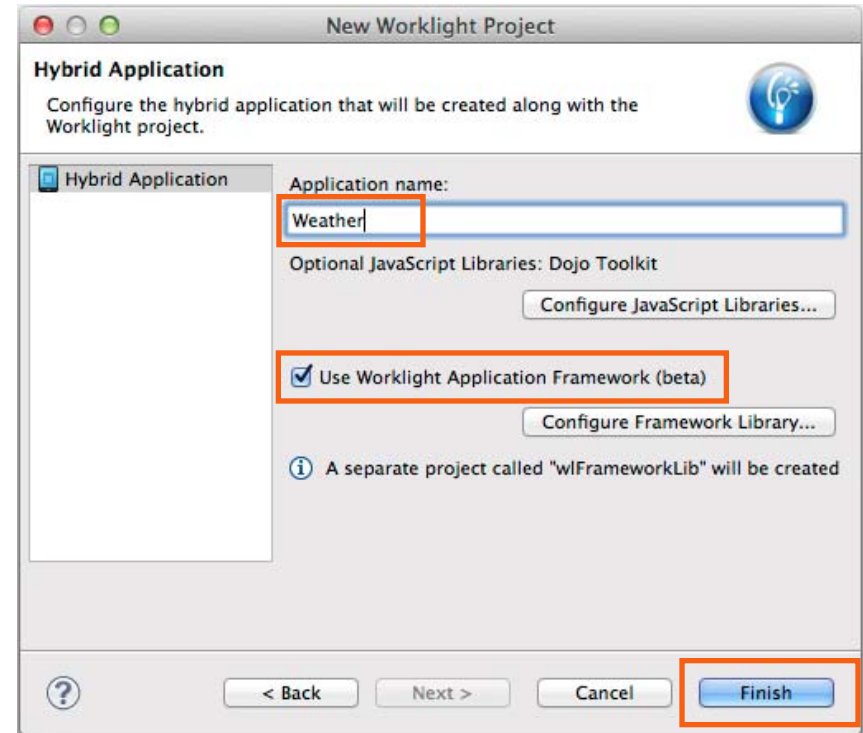
# New Worklight Project

- In **Name**, type WeatherProject.
- In **Project Templates**, select **Hybrid Application**.
- Click **Next**.



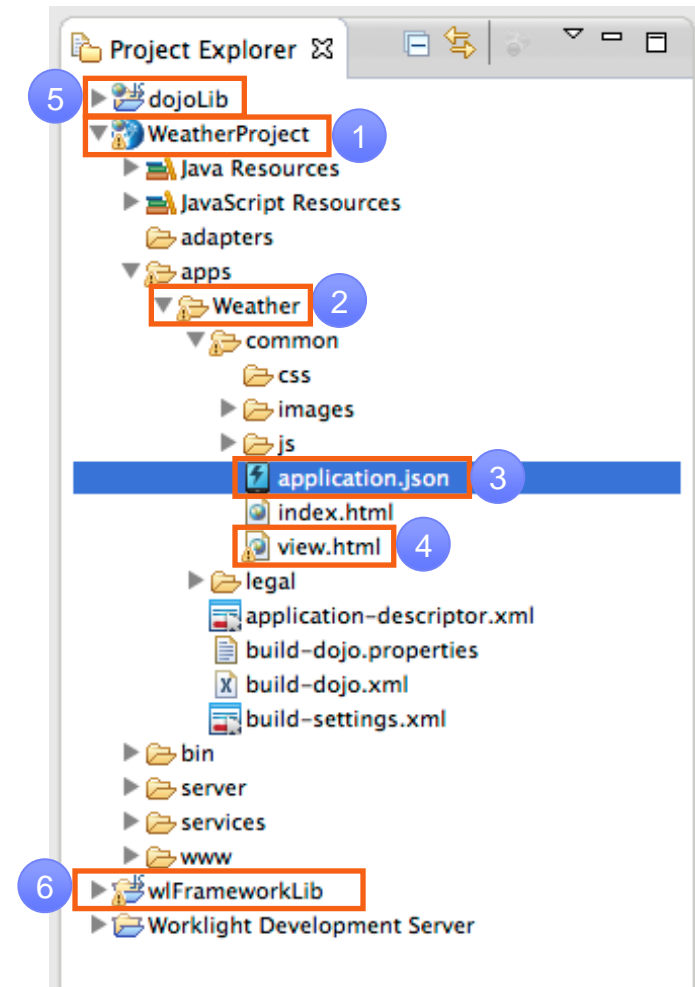
# New Hybrid Application

- In **Application name**, type Weather.
- Select **Use Worklight Application Framework (beta)**.
  - **Dojo Toolkit** is selected automatically because IBM Worklight Application Framework is based on Dojo.
- Click **Finish**.
- If the **Open Associated Perspective** dialog box opens, click **No**.



## What Was Created in the Workspace?

- Worklight project **WeatherProject** (1).
  - Hybrid application **Weather** (2).
    - **application.json** file: the application model (3).
    - **view.html** file: the application UI (4).
- **dojoLib** project: Dojo Toolkit library (5).
- **wlFrameworkLib** project: IBM Worklight Application Framework library (6).



## *The IBM Worklight Application Framework Editor*

- When you create a new hybrid application that uses IBM Worklight Application Framework, the editor of IBM Worklight Application Framework opens.
- Use this editor to edit the `application.json` and the `view.html` files of your app.
- If you close the editor, double-click the `application.json` file to open it again.
- The editor has three tabs (see the next slide):
  - **Application** is used to specify the general configurations of your app.
  - **Data Objects** is used to define the data objects of your app, and how the data maps to the back-end services.
  - **Views** is used to define your app views (screens) and the transitions between them.



# The IBM Worklight Application Framework Editor

application.json

Worklight Application Framework editor

### Getting started with Worklight Application Framework

Create Data Object from Service

Create a View

### Application Level Events

Configure application-level events, for instance when the application starts, or is about to go into the background.

Event	Function	File

Add...  
Remove  
Edit

**Theme**  
Customize the application UI using CSS

Theme:

**Authentication**

User Login is required:

Realm:

Automatically re-authenticate after expired session

Details:

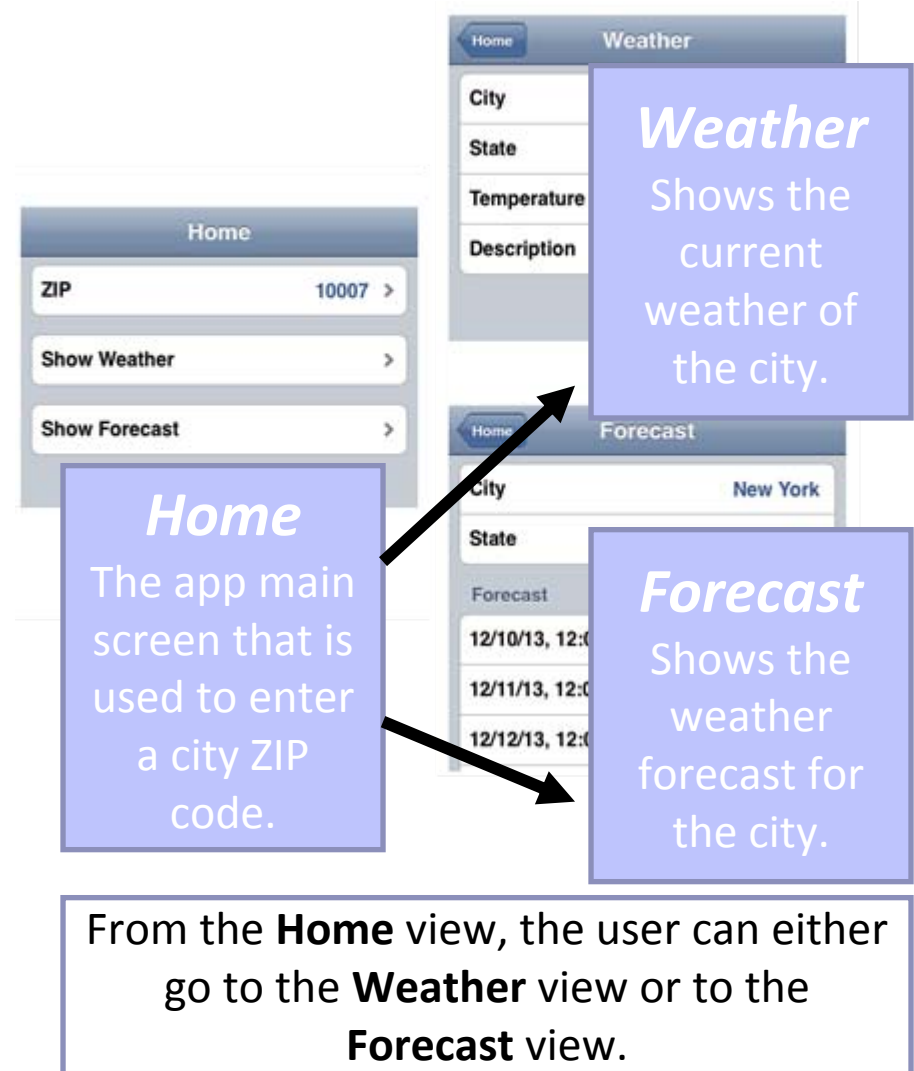
Application | Data Objects | Views

# Agenda

- Creating a project with a hybrid application
- **Weather application overview**
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- Deploying and previewing

# Weather Application Overview

- You build a `Weather` app.
- The app has three views (screens), as shown on the right.
- The **Home** view does not access any back-end service. This view is used to enter the ZIP code of a city.
- The **Weather** and **Forecast** views pull the data for the requested ZIP code from back-end services.
- The app uses a public weather Web Service:  
[http://wiki.cdyne.com/index.php/C\\_DYNE\\_Weather](http://wiki.cdyne.com/index.php/C_DYNE_Weather)

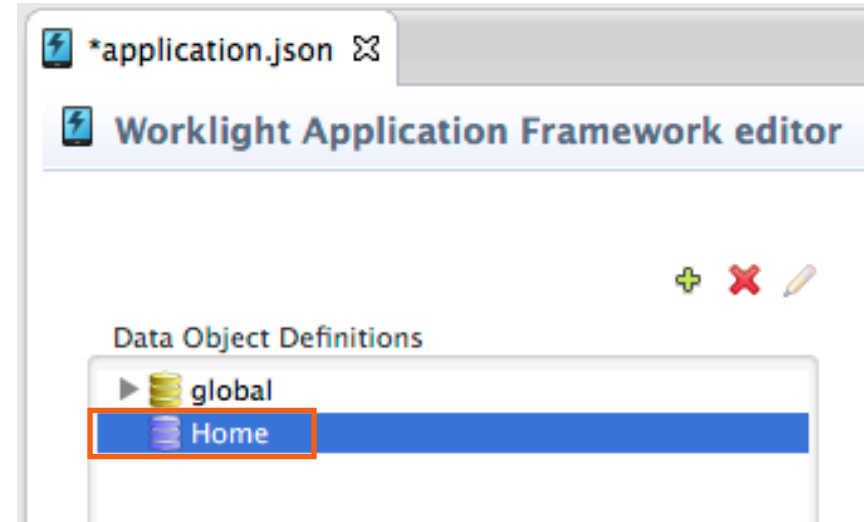


# Agenda

- Creating a project with a hybrid application
- Weather application overview
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- Deploying and previewing

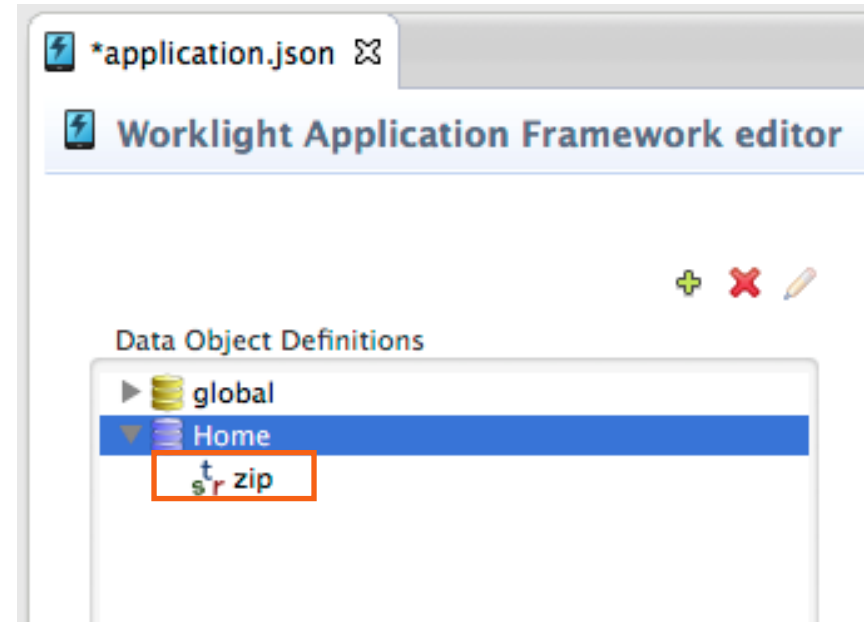
## Data Object of the Home View

- Create a data model and a view.
  - You can create the view first, or you can create the data model first. In this tutorial you create the data model first.
- Switch to the **Data Objects** tab.
- Click **DataObject** (created by default).
- Press F2 and rename the data object Home.



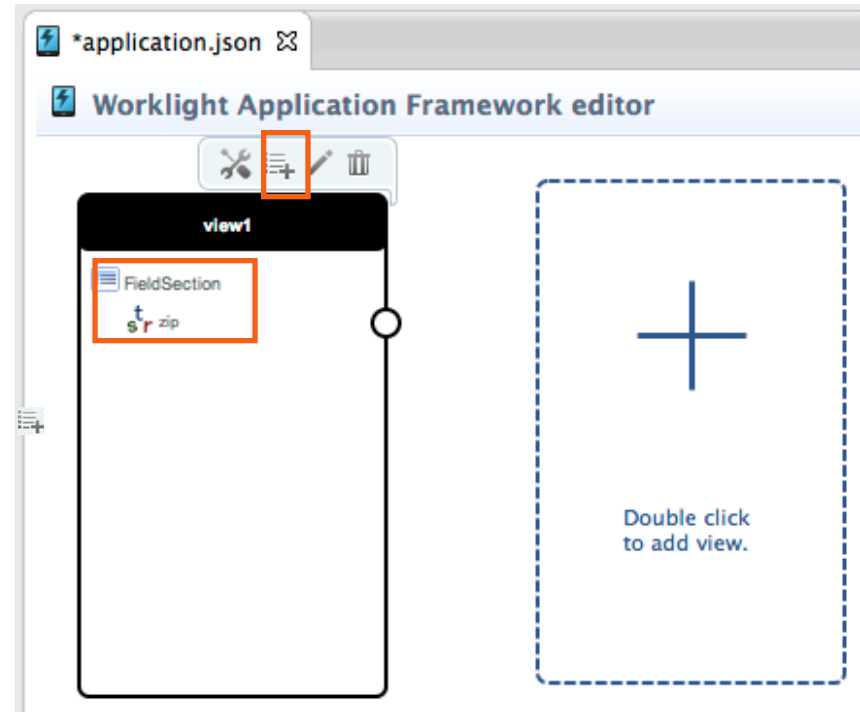
## Adding a Data Attribute

- Create a data attribute to store the ZIP code that the user of the app will enter.
- Right-click **Home** data object, and select **New**. The “New field” window opens.
- In **Name**, enter: `zip`
- Select **String**, and click **OK**.



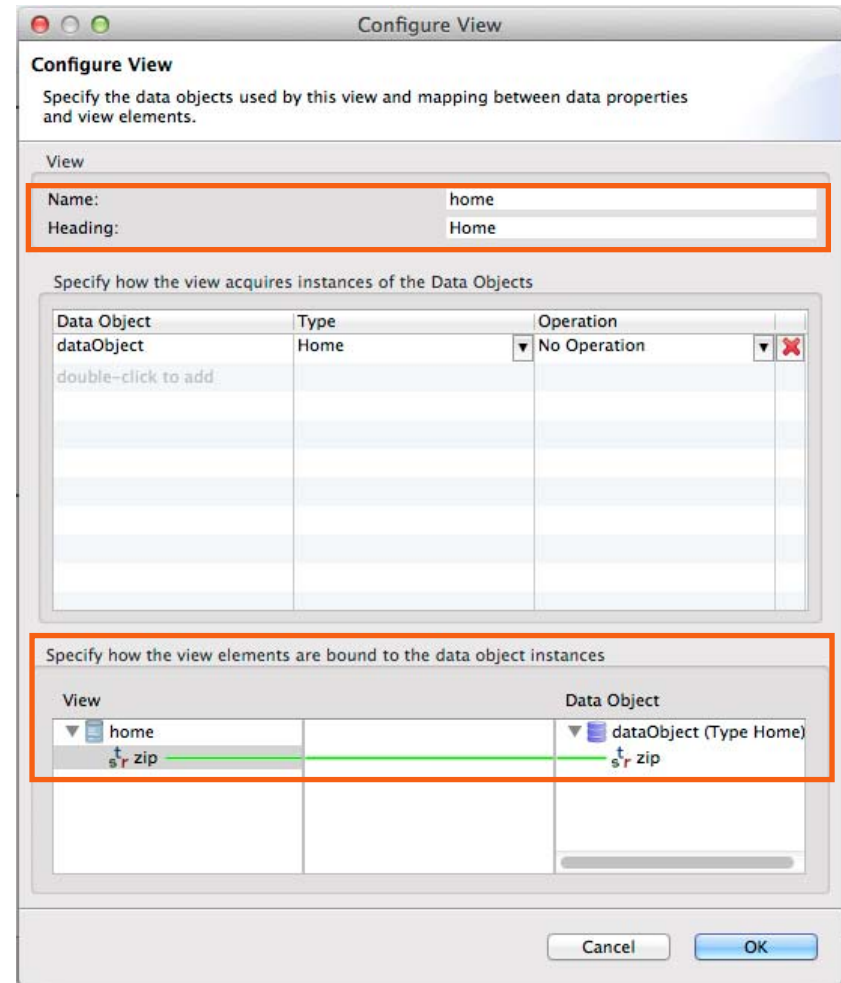
# Creating the Home View

- Switch to the **Views** tab.
- Create an editable view element to show the ZIP code. View elements that are not tables must be placed in sections.
- Hover over **view1** (created by default). A View menu is displayed over the view.
- Click **Add View Element**. In the **Add View Element** window, select **Section**, and click **OK**.
- Right-click **FieldSection**, and click **Add View Element**.
- In the **Name** field, type `zip` and in the **Label** field, type `ZIP`. Select **String**, and click **OK**.
- Double-click the new **zip** element, select **Editable**, and click **OK**.
- Note:** Names are used to identify entities in the editor, labels are displayed in the application UI.



## Creating the Home View (continued)

- Hover over **view1**. The View menu is displayed over the view.
- Click **Configure View** (the pencil icon) to open the **Configure View** window.
- In **Name**, type `home`, and in **Heading**, type `Home`.
- In **Specify how the view elements are bound to the data object instances** at the bottom of the window, drag **zip** from the **View** side (left) to **zip** on the **Data Object** side (right) to perform the binding.
- Click **OK** to close the window.



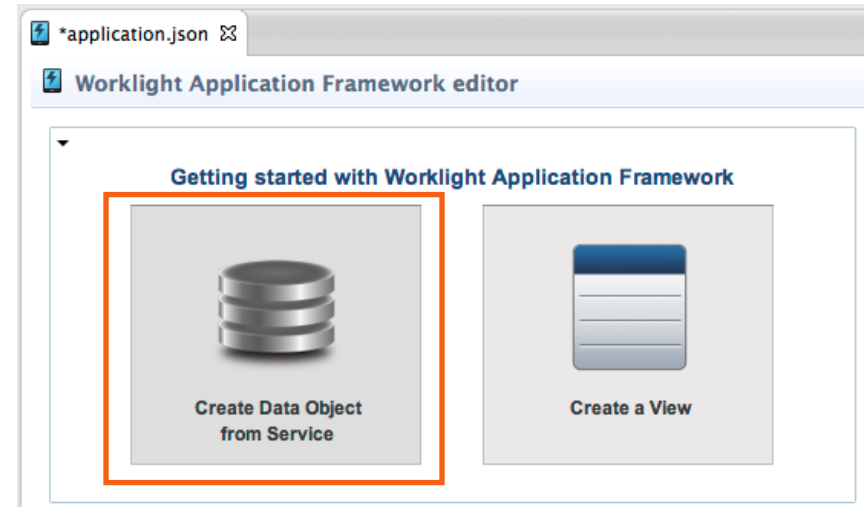


# Agenda

- Creating a project with a hybrid application
- Weather application overview
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- Deploying and previewing

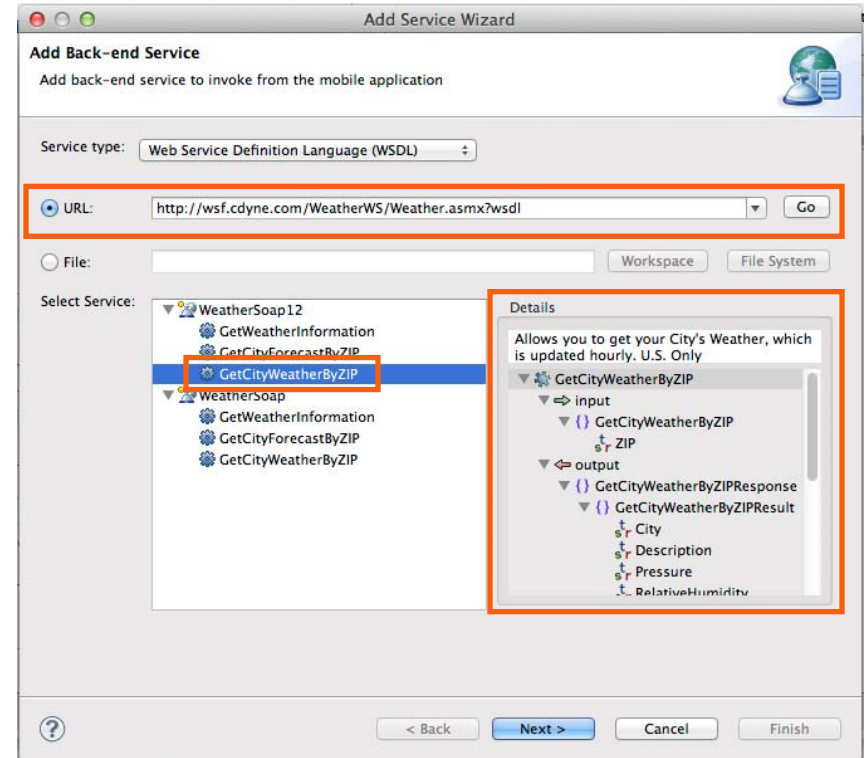
## Creating the Weather View

- Create the **Weather** view based on an existing web service.
- Switch to the **Application** tab and click **Create Data Object from Service**.



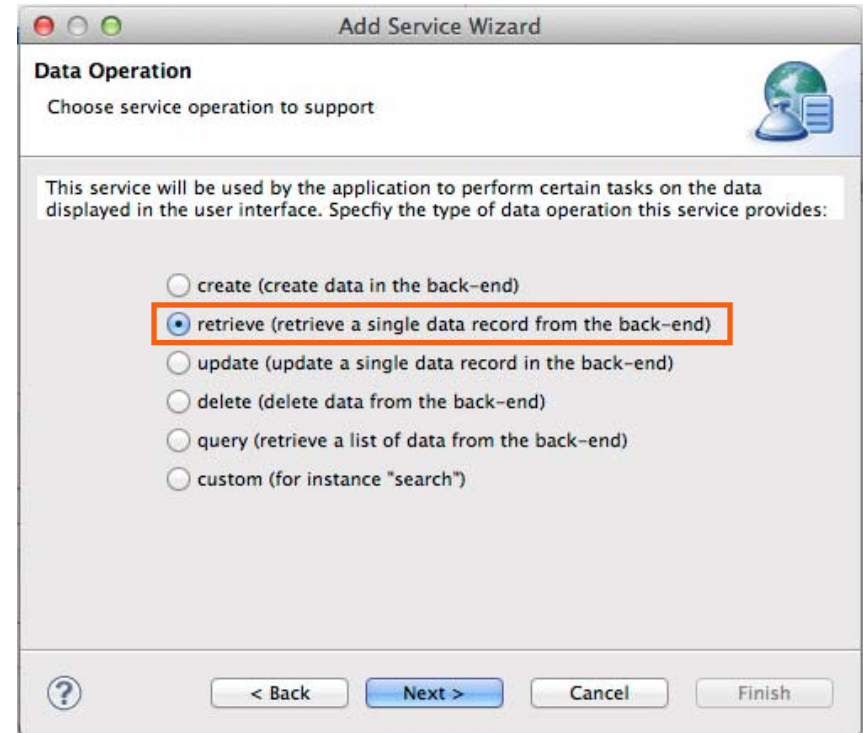
# Service Discovery

- In the **Service type** list, select **Web Service Definition Language (WSDL)**.
- Copy the following URL, and paste it into the **URL** field:  
`http://wsf.cdyne.com/WeatherWS/Weather.asmx?wsdl`  
 Click **Go**.
- In **Select Service** (lower left), expand **WeatherSoap12**, and select **GetCityWeatherByZIP**.
- Browse the **Details** section (lower right) to find out more about the service.
- Click **Next**.



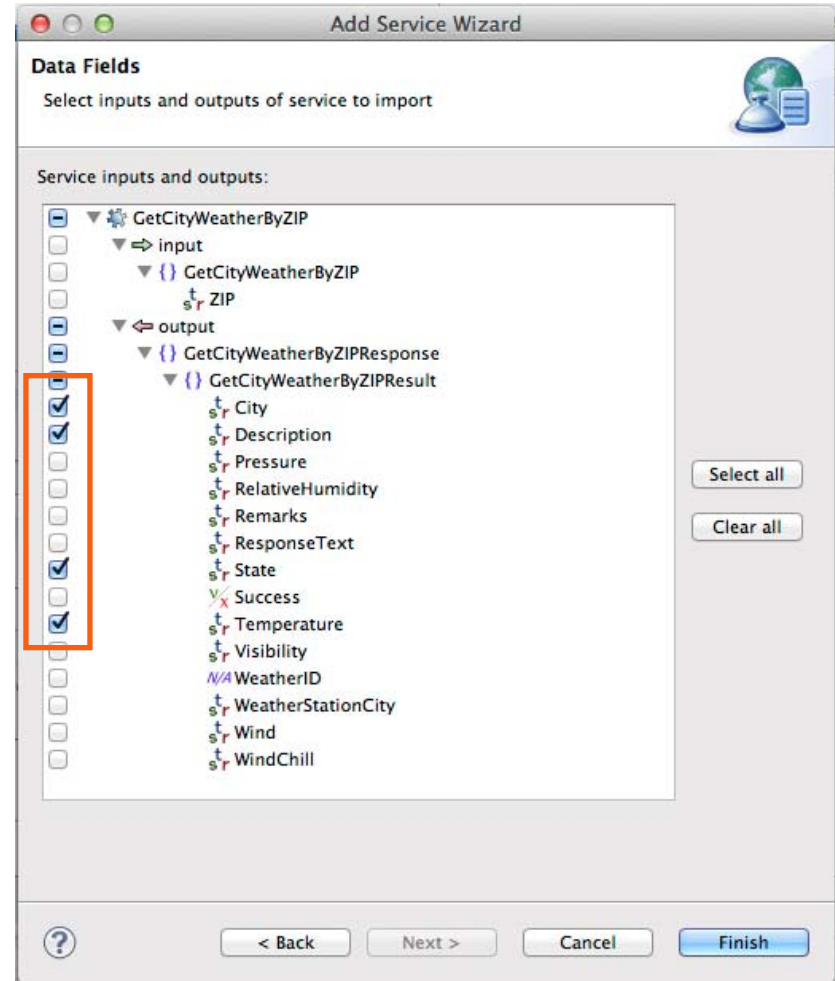
# Data Operation

- In addition to attributes, data objects have operations. The application uses operations to interact with back-end services via adapters. Operations define the mappings between the data attributes and the service inputs and outputs.
- Select **retrieve (retrieve a single data record from the back-end)** and click **Next**.



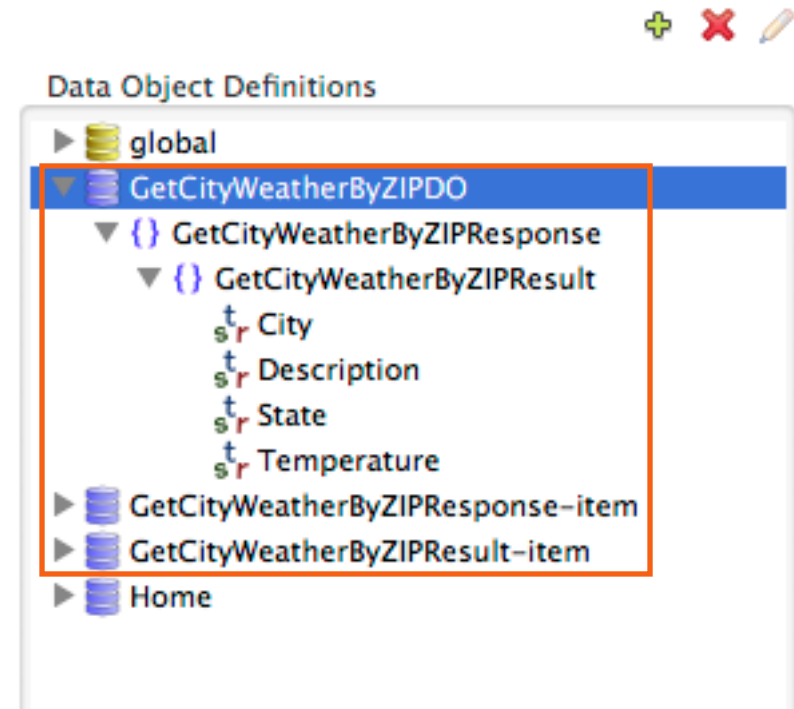
# Data Attributes

- Define the data attributes to create, based on the service inputs and outputs.
- Select **City**, **Description**, **State**, and **Temperature** under **output** (as shown on the right).
- Click **Finish**.



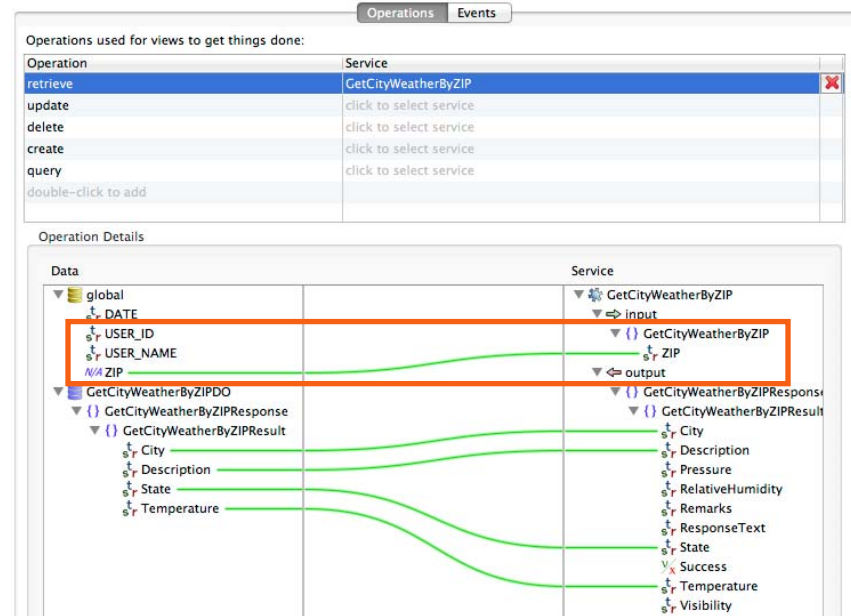
## Created Data Objects

- The editor switches to the **Data Objects** tab.
- Based on your selections, three data objects were created to replicate the structure of the **GetCityWeatherByZIP** service.
- The root data object is **GetCityWeatherByZIPDO**.



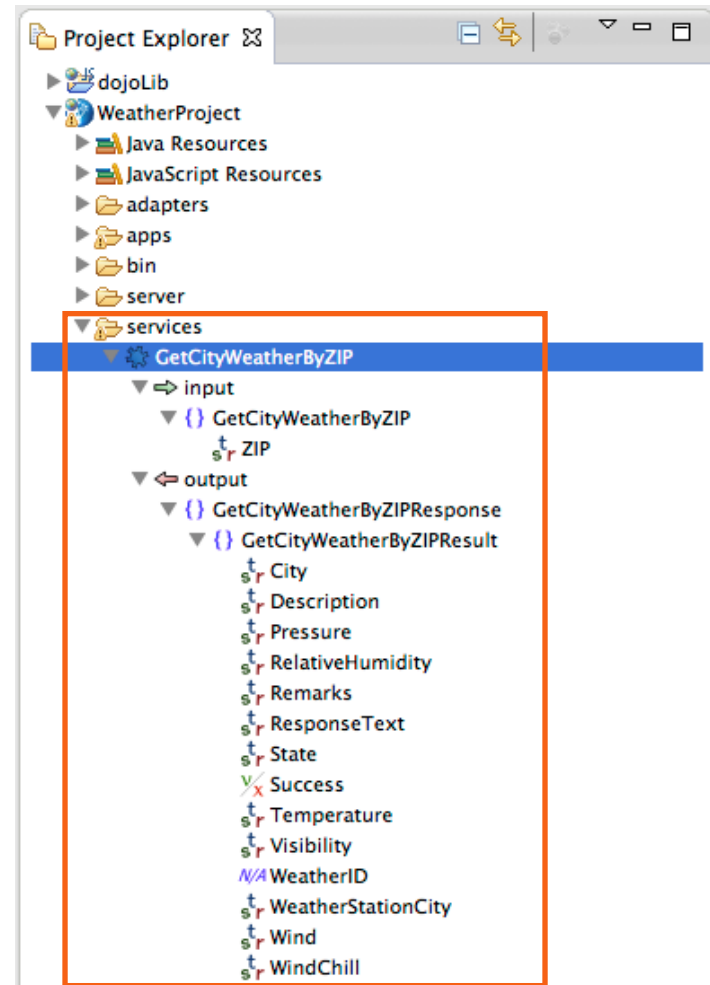
# Created Operation

- According to your selection, a **retrieve** operation is associated with the **GetCityWeatherByZIPDO** data object.
- The service outputs that you selected previously are mapped to the data attributes.
- The service requires **ZIP** as input. Create a global variable and map it to the service input: drag **ZIP** from the **Service** (right) side to **global** on the **Data** (left) side.
- You will later assign the ZIP code entered by the user to this global variable.



## Discovered Service

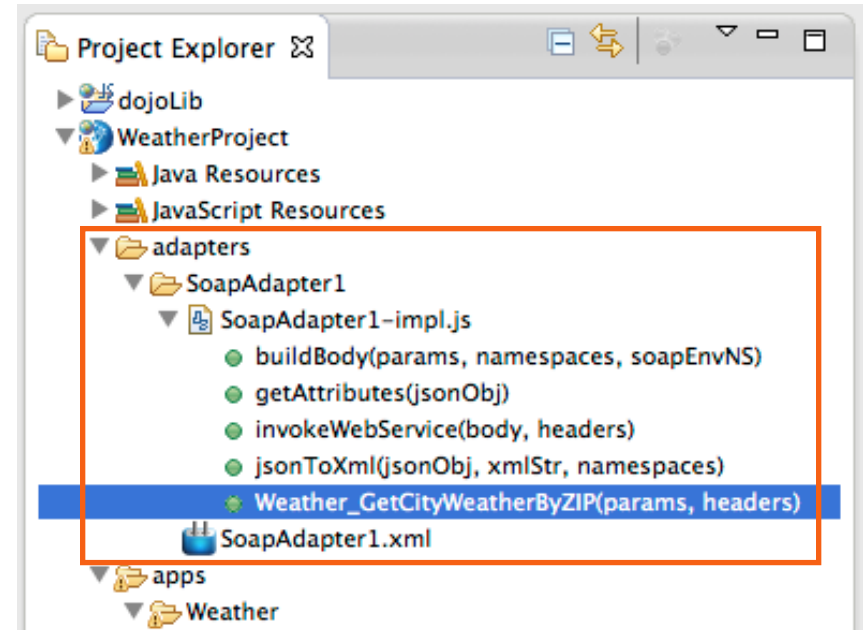
- The discovered services are in the **services** folder of the Worklight project.
- Expand the **services** folder to see the **GetCityWeatherByZIP** service.





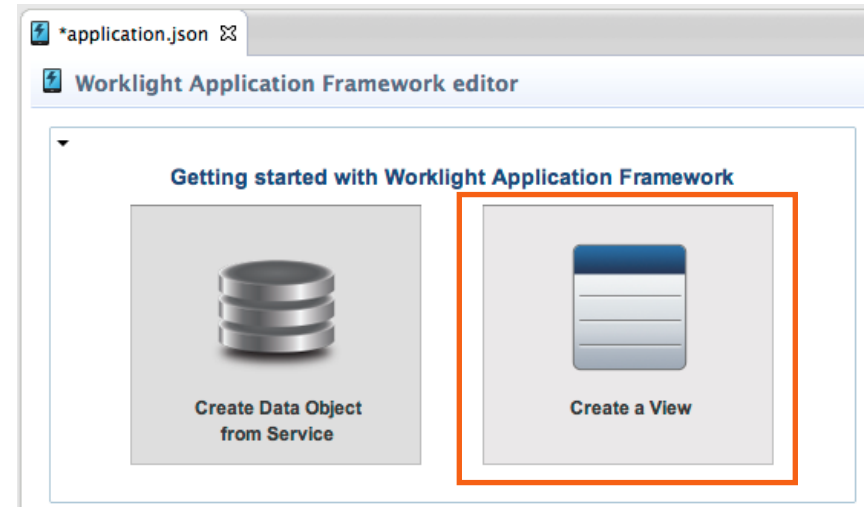
## Generated Adapter

- To invoke the back-end service, an adapter was automatically generated for your application.
- The new adapter is in the **adapters** folder. The name of the adapter is **SoapAdapter1**.
- This adapter is used whenever the application needs to access back-end services.
- In this particular case, you set up the **Weather** view to call the data object **retrieve** operation when the view opens. The operation invokes the **Weather\_GetCityWeatherByZIP** procedure of this adapter.



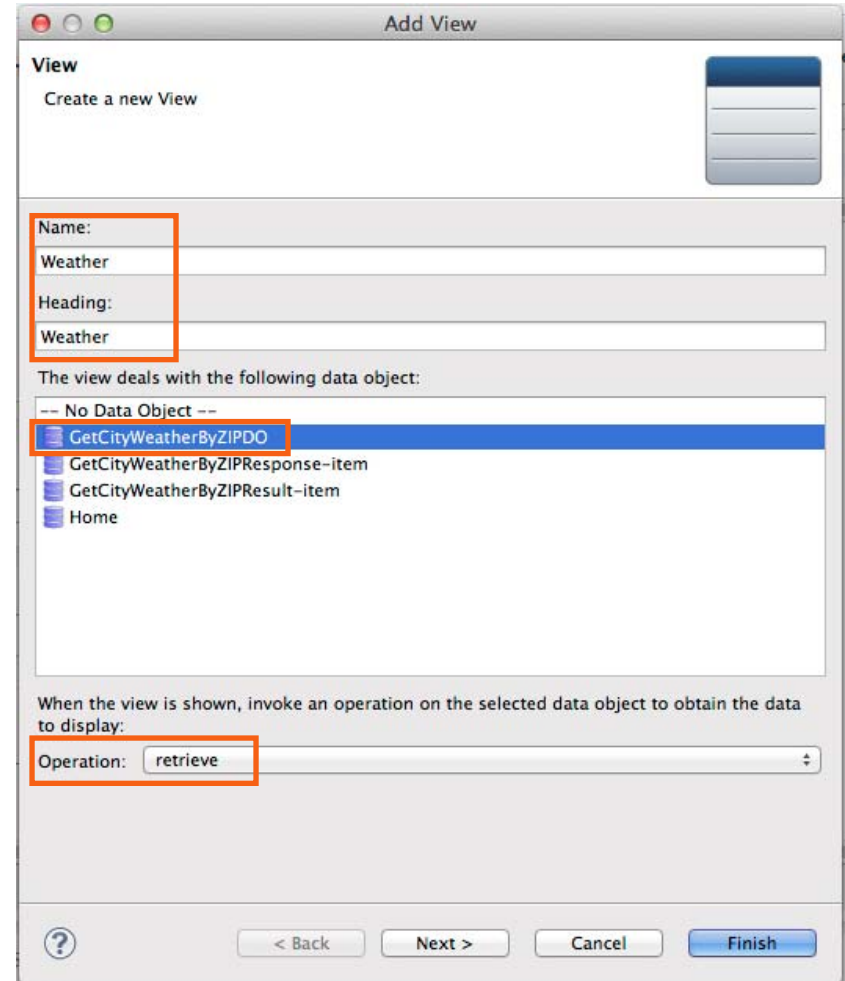
## Creating the Weather View

- You discovered a service and created a data object based on its outputs.
- Now, create a view by using this data object as a template.
- Go back to the **Application** tab and click **Create a View**.



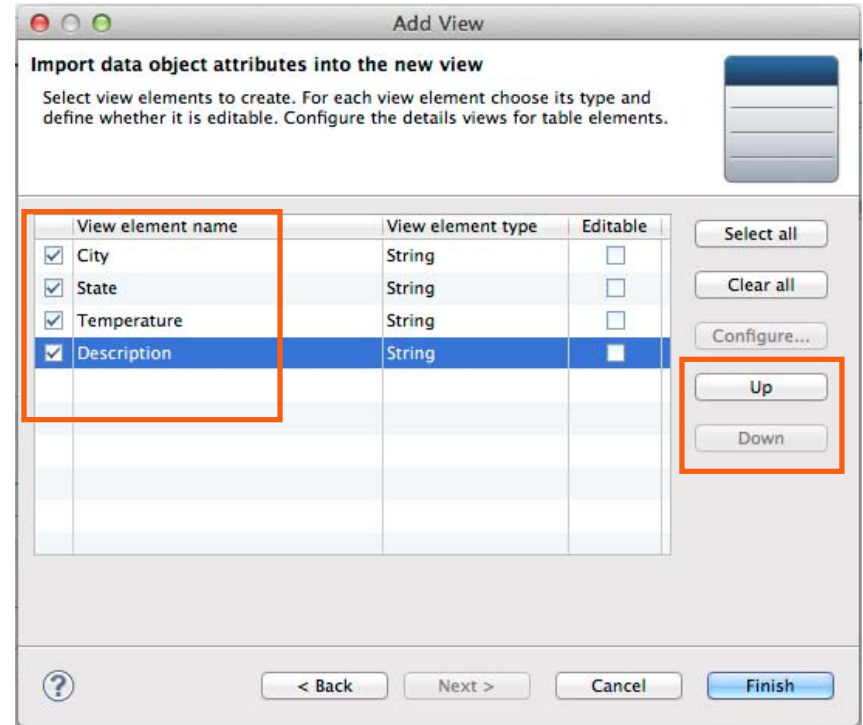
# Creating the Weather View

- In **Name** and **Heading**, type: Weather.
- Select **GetCityWeatherByZIPDO**. The view represents this data object. The view elements are created based on the attributes of this data object.
- Select **retrieve** from the **Operation** list, and click **Next**. This operation is called when the view opens.



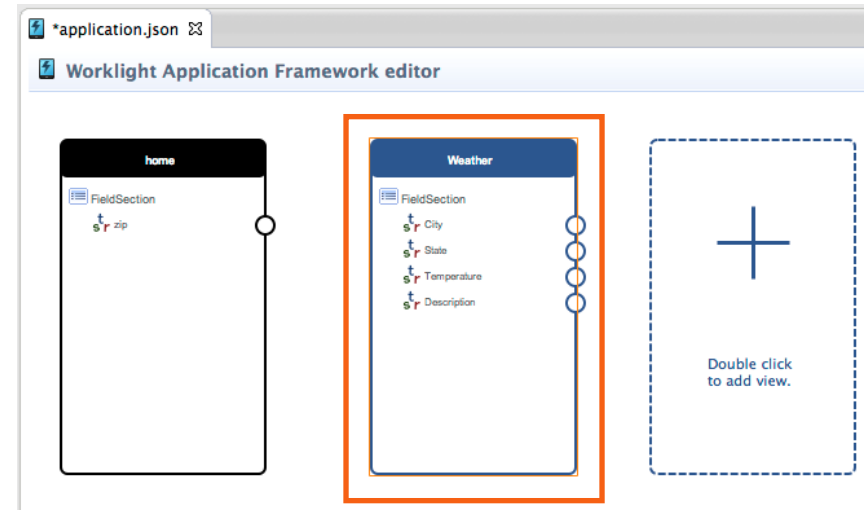
# Configuring the View Elements

- You create the view elements that are listed in the table.
- You can deselect elements that you do not want to create by clearing check boxes in the first column. For this view, create all elements.
- Reorder the elements as follows by clicking **Up** or **Down**: **City, State, Temperature, Description**.
- Click **Finish**.



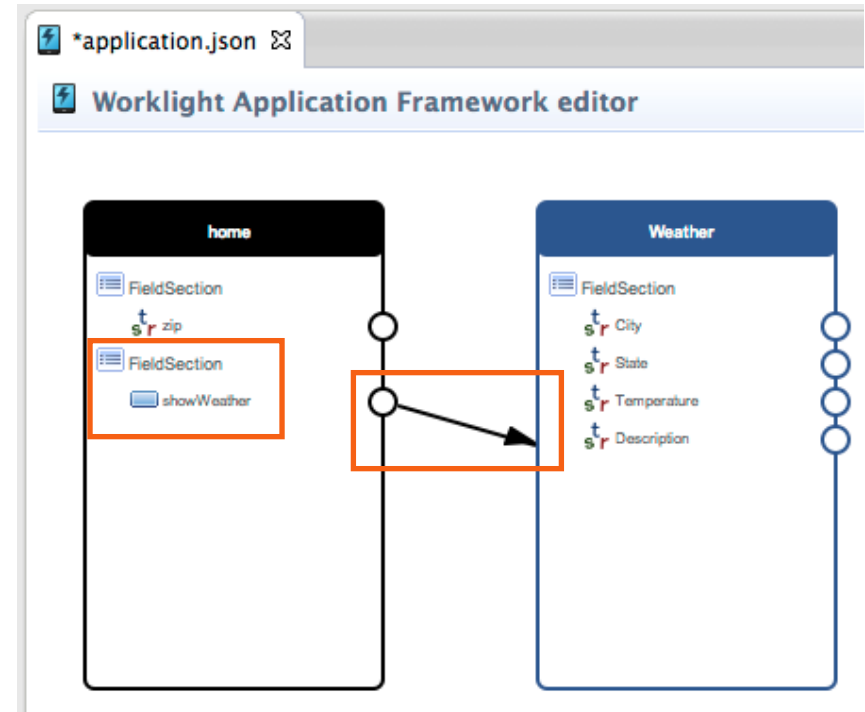
## Created View

- Switch to the **Views** tab. You can see the **Weather** view that you created.
- The view has a section that contains the elements that you selected.



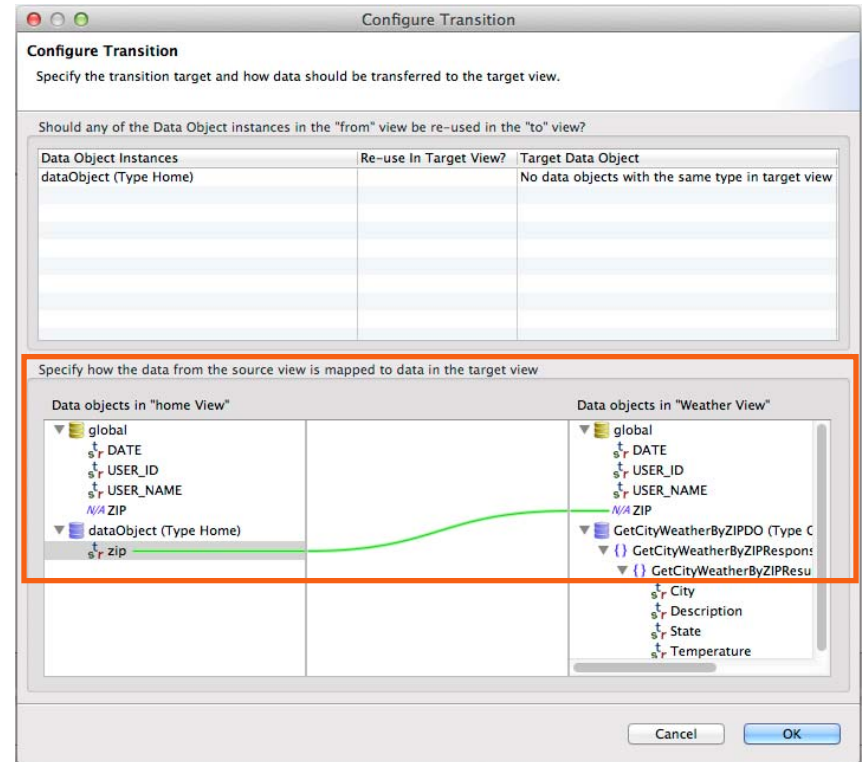
## Creating a Transition from the Home View to the Weather View

- First, create a button that causes the transition.
- Hover over the **home** view, click **Add View Element**, select **Section**, and click **OK**.
- Right-click the new **FieldSection** (the second one in the view) and click **Add View Element**.
- In the **Name** field, type `showWeather`, and in the **Label** field, type `Show Weather`. Select **Button**, and click **OK**.
- Create a transition from the new **showWeather** button, by dragging the circle in front of **showWeather**, on the border of the view, to the **Weather** view.



# Configuring the Transition

- Double-click the transition to configure it. The **Configure Transition** window opens.
- At the bottom of the window, you can see the **Specify how the data from the source view is mapped to data in the target view** section.
- Map (drag) the **zip** attribute of the source view data object (on the left) to the global variable **ZIP** (on the right) that was created earlier.
- When the transition happens, the value of the **zip** attribute is assigned to the global variable **ZIP**.
- Click **OK** to close the window.



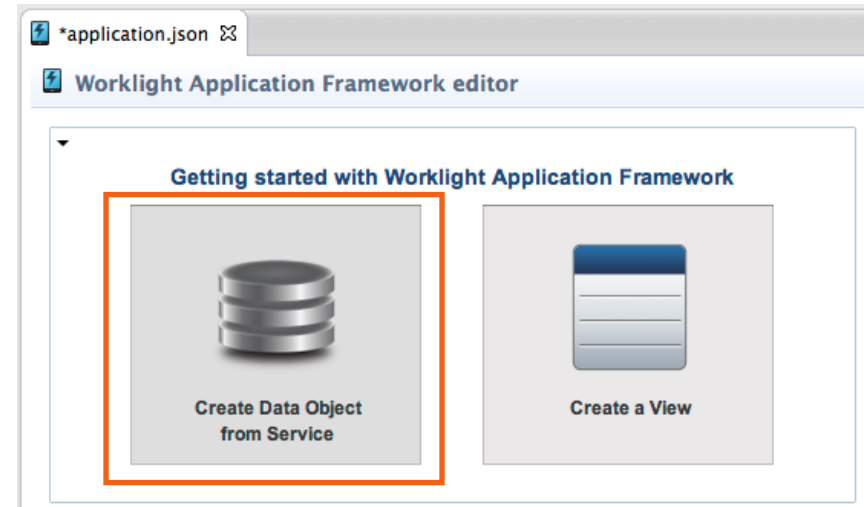
# Agenda

- Creating a project with a hybrid application
- Weather application overview
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- Deploying and previewing



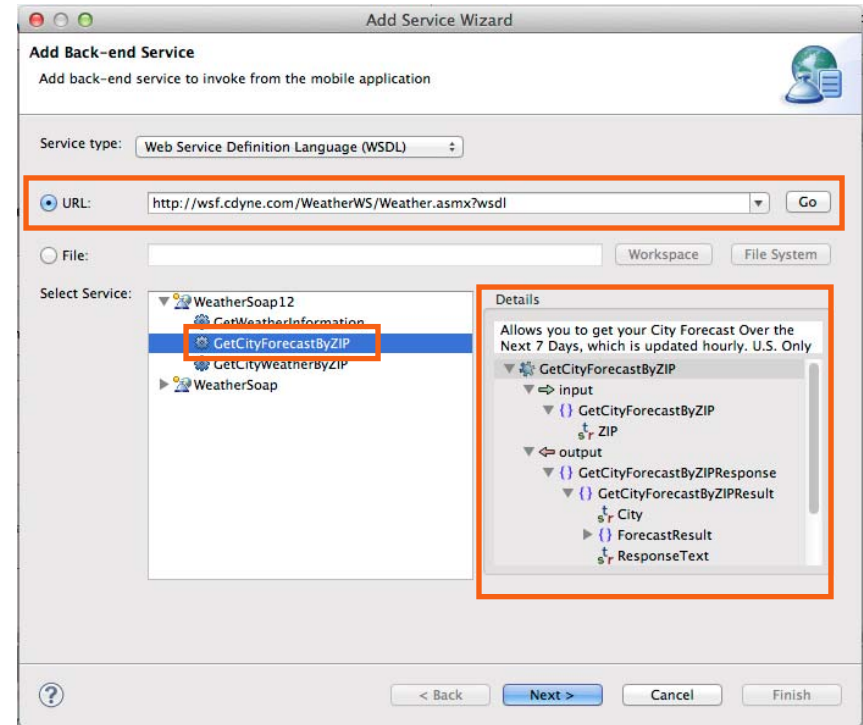
## Creating the Forecast View

- Switch to the **Application** tab and click **Create Data Object from Service**.



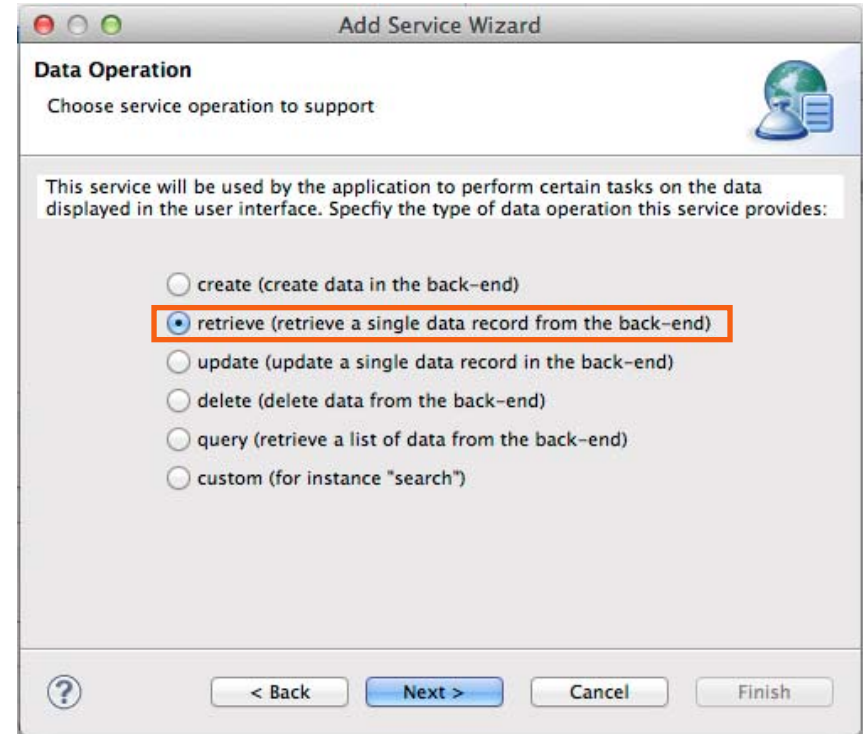
# Creating the Forecast View – Service Discovery

- In **Service type**, select **Web Service Definition Language (WSDL)**.
- Select the URL you used previously in the **URL** list, and click **Go**.
- In the **Select Service** section (lower left), expand **WeatherSoap12**, and select **GetCityForecastByZIP**.
- Browse the **Details** section (lower right) to find out more about the service.
- Click **Next**.



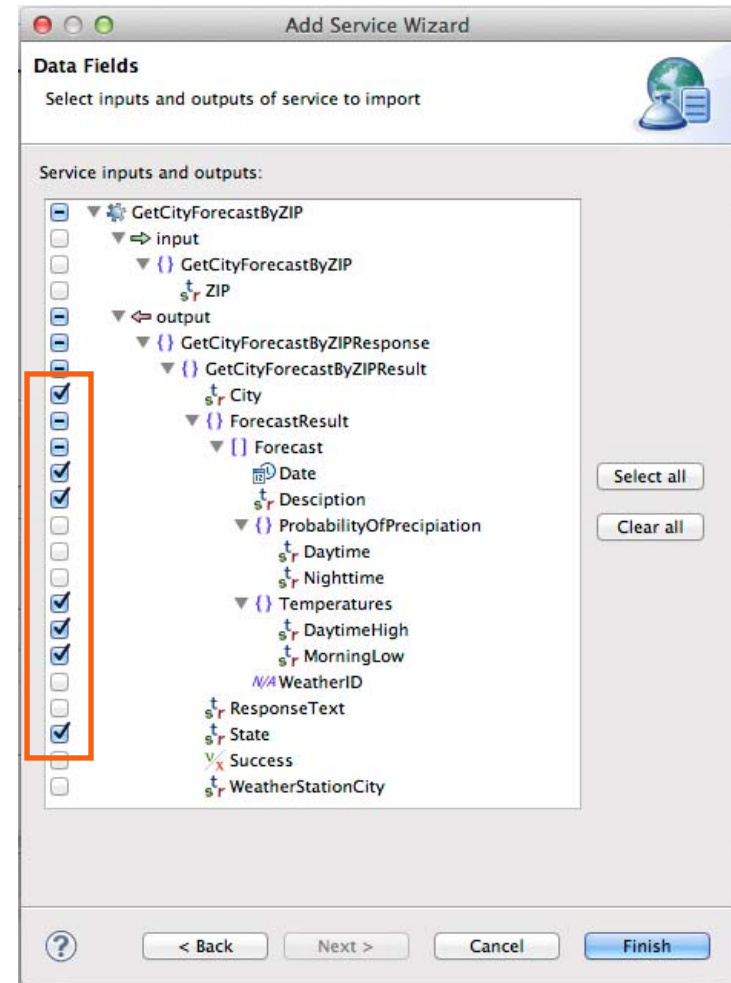
# Data Operation

- Select the data operation that is used to invoke the service.
- Select **retrieve (retrieve a single data record from the back-end)**, and click **Next**.



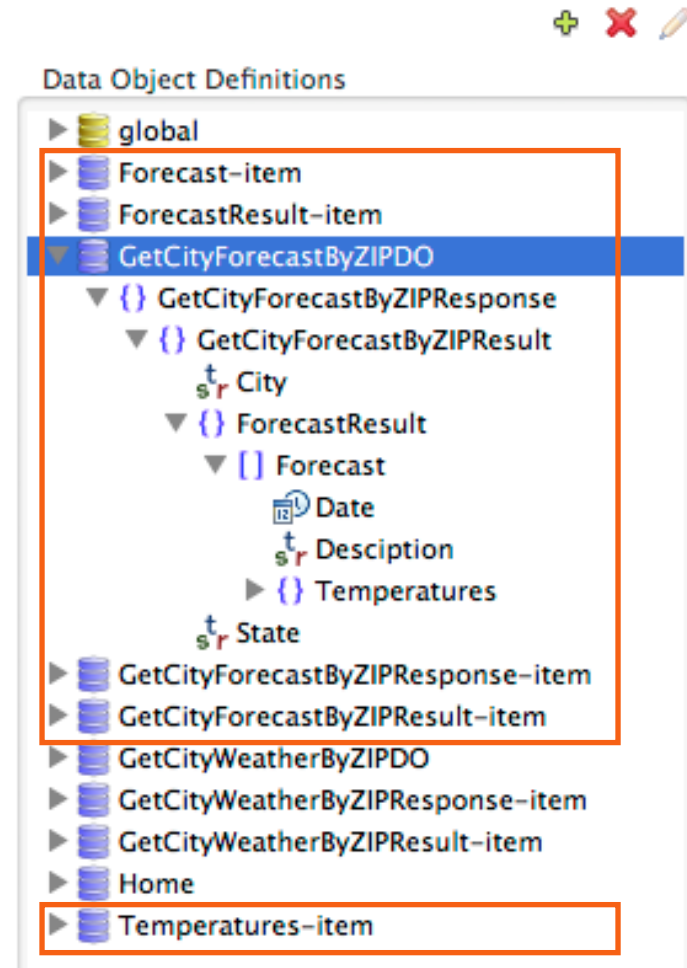
# Data Attributes

- Define the data attributes to create, based on the service inputs and outputs.
- Select **City**, **Date**, **Description**, **DaytimeHigh**, **MorningLow**, and **State** under **output** (as shown on the right).
- Click **Finish**.



## Created Data Objects

- The editor switches to the **Data Objects** tab.
- Based on your selections, six data objects were created to replicate the structure of the **GetCityForecastByZIP** service.
- The root data object is **GetCityForecastByZIPDO**.



# Created Operation

- According to your selection, a **retrieve** operation is associated with the **GetCityWeatherByZIPDO** data object.
- The service outputs that you selected previously are mapped to the data attributes.
- The service requires **ZIP** as an input. Drag **ZIP** from the **Service** (right) side to **ZIP** on the **Data** (left) side, to map the global variable to the service input.

Operations used for views to get things done:

Operation	Service
retrieve	GetCityForecastByZIP
update	click to select service
delete	click to select service
create	click to select service
query	click to select service
double-click to add	

Operation Details

Data

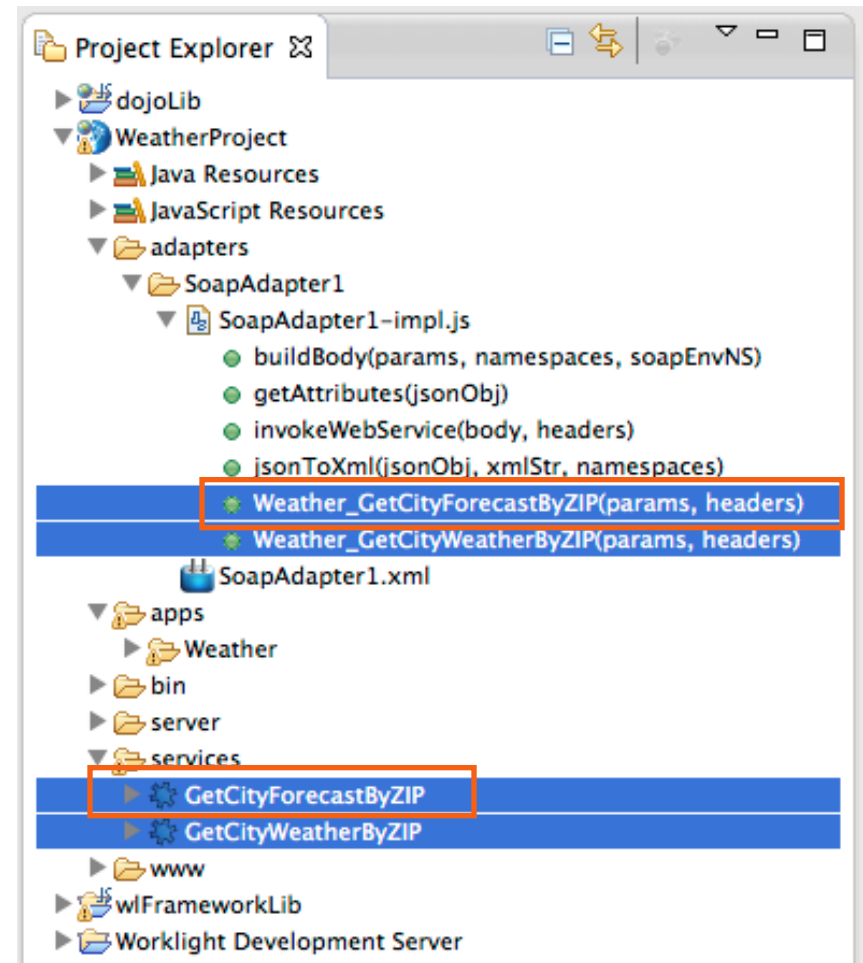
- global
  - DATE
  - USER\_ID
  - USER\_NAME
  - ZIP
- GetCityForecastByZIPDO
  - GetCityForecastByZIPResponse
    - GetCityForecastByZIPResult
      - City
      - ForecastResult
        - Forecast
          - Date
          - Description
          - Temperatures
            - DaytimeHigh
            - MorningLow
          - State

Service

- GetCityForecastByZIP
  - Input
    - GetCityForecastByZIP
      - ZIP
  - output
    - GetCityForecastByZIPRespo
      - GetCityForecastByZIPRes
        - City
        - ForecastResult
          - Forecast
            - Date
            - Description
            - ProbabilityOfPr
              - Daytime
              - Nighttime
            - Temperatures
              - DaytimeHigh
              - MorningLow
            - WeatherID
            - ResponseText
            - State
            - Success

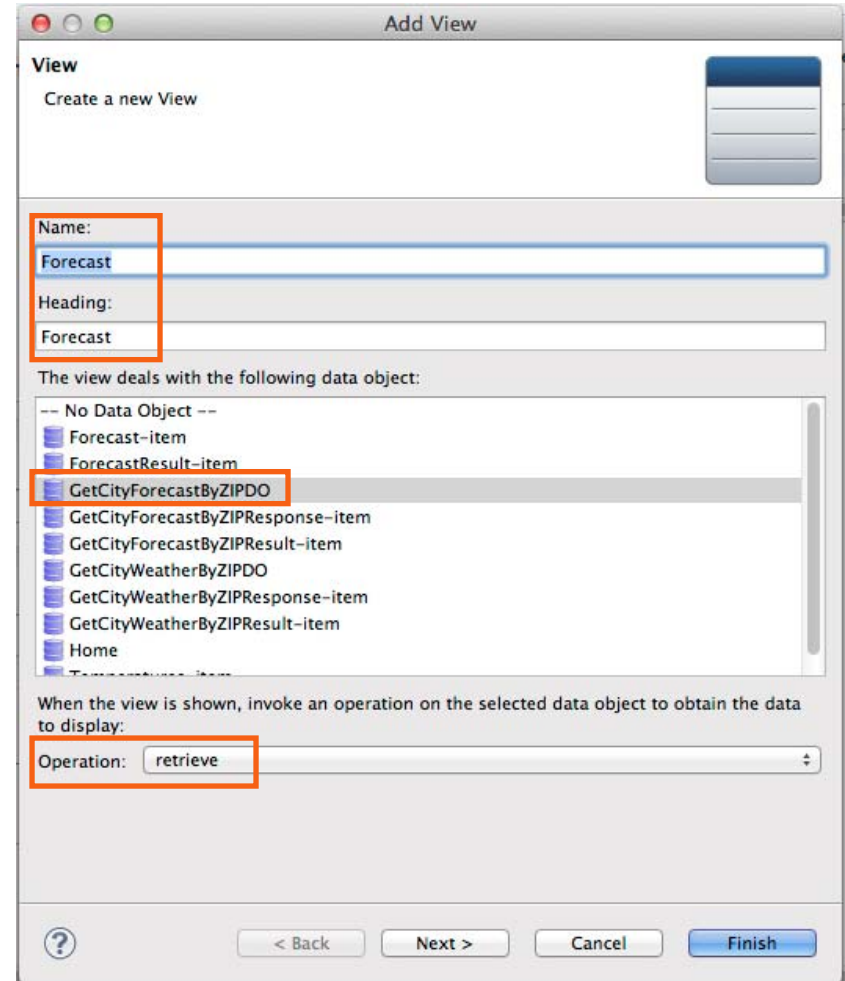
## Discovered Service and Generated Adapter Procedure

- The discovered **GetCityForecastByZIP** service is placed in the **services** folder of the Worklight project.
- To invoke the back-end service, a new procedure is generated in **SoapAdapter1**.



# Creating the Forecast View

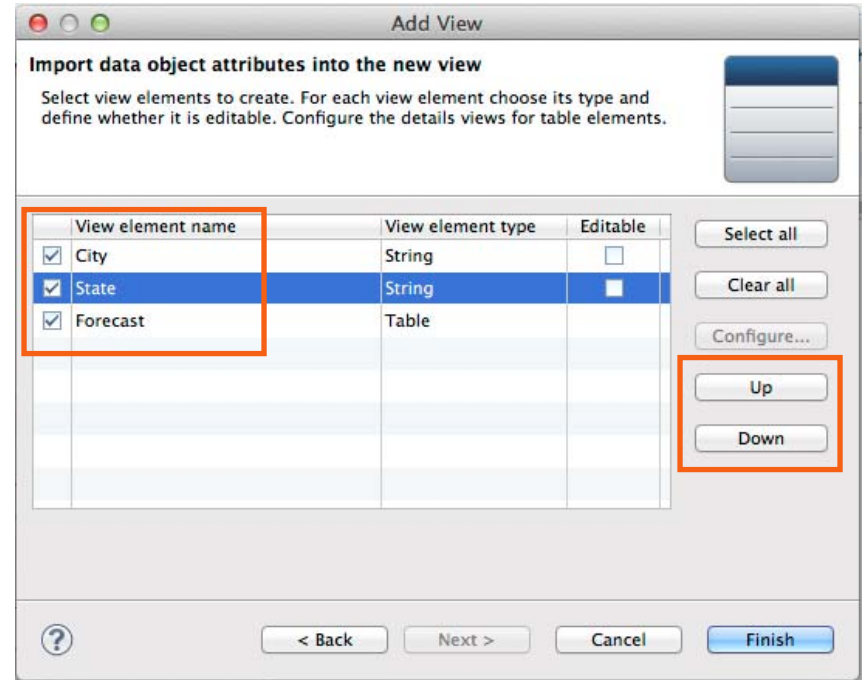
- Go back to the **Application** tab and click **Create a View**.
- In **Name** and **Heading**, type Forecast.
- Select **GetCityForecastByZIPDO**.
- Select **retrieve** in the **Operation** list, and click **Next**.





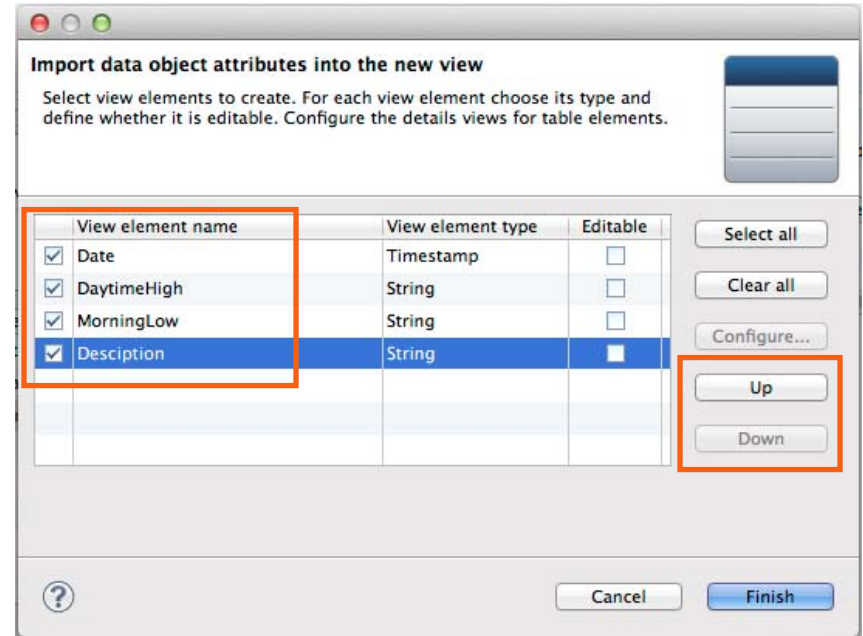
# Configuring the View Elements

- Reorder the elements as follows by clicking **Up** or **Down**: **City**, **State**, **Forecast**.



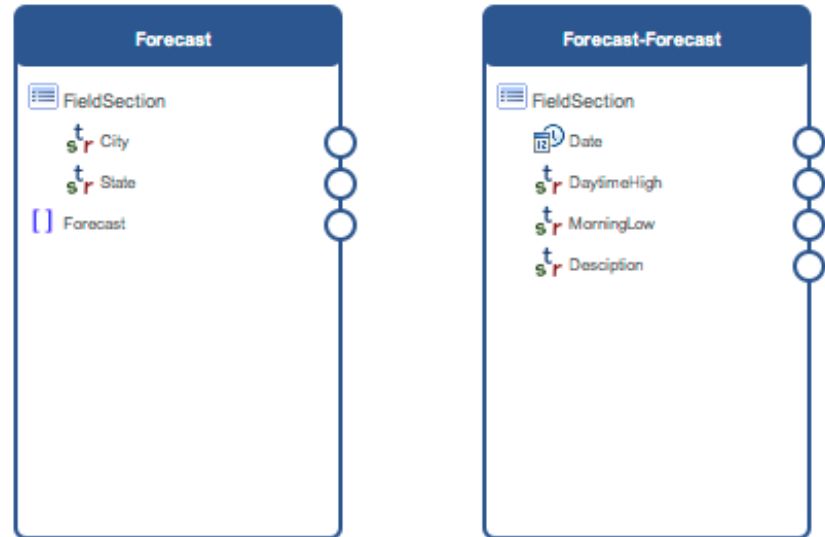
# Configuring the Table Element

- Click the **Forecast** row, and click **Configure**.
- Reorder the elements as follows by clicking **Up** or **Down**: **Date**, **DaytimeHigh**, **MorningLow**, **Description**.
- Click **Finish** to close the window.
- Click **Finish** to close the **Add View** wizard.



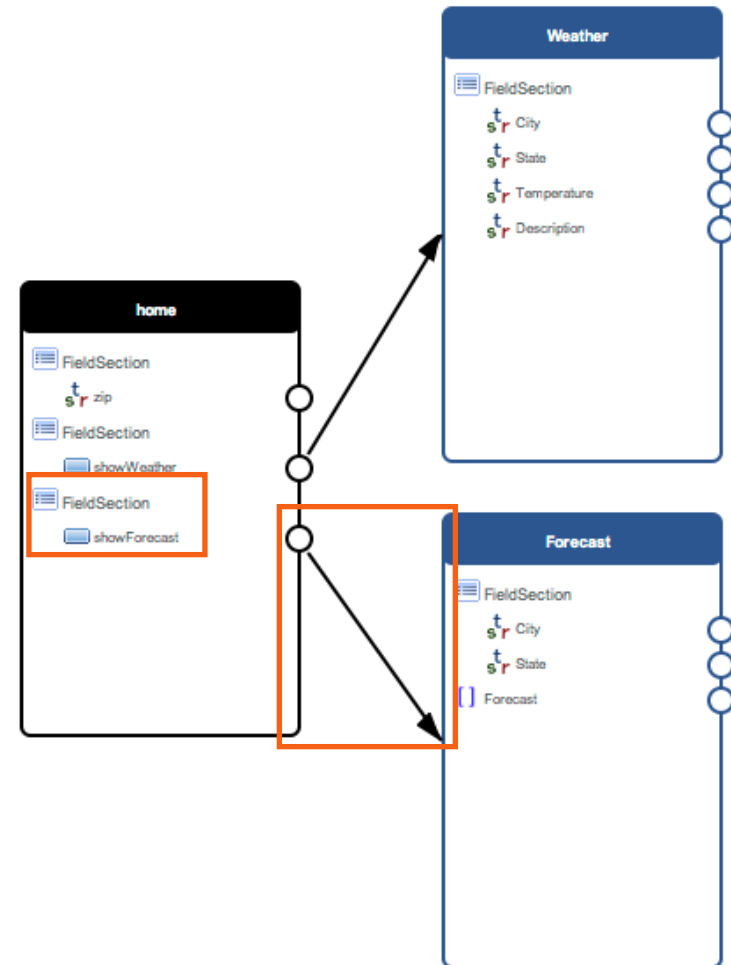
## Created Views

- Switch to the **Views** tab to see what was created.
- The **Forecast** view shows **City**, **State**, and the **Forecast** table.
- The **Forecast** table has four attributes. On a small device, the table can show only two columns.
- To show all four attributes, another view was created – **Forecast-Forecast** (the concatenation of the name of the first view and the field name of its table).
- This **Forecast-Forecast** view opens when you tap the row of the table in the running application.



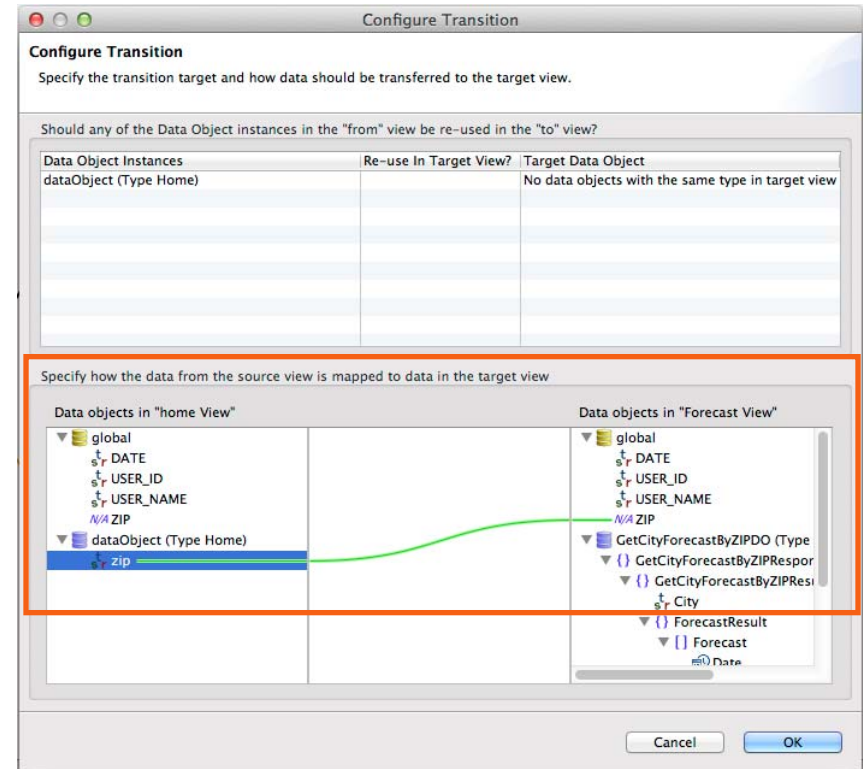
# Creating a Transition from Home View to Forecast View

- First, create a button that causes the transition.
- Hover over **home** view, click **Add View Element**, select **Section**, and click **OK**.
- Right-click the new **FieldSection** (the third one in the view) and click **Add View Element**.
- In the **Name** field, type `showForecast`, and in the **Label** field, type `Show Forecast`. Select **Button**, and click **OK**.
- Create a transition from the new **showForecast** button, by dragging the circle in front of **showForecast**, on the border of the view, to the **Forecast** view.



# Configure Transition

- Double-click the created transition to configure it. The **Configure Transition** window opens.
- At the bottom of the window, you can see the **Specify how the data from the source view is mapped to data in the target view** section.
- Map (drag) the **zip** attribute of the data object in the source view (on the left) to the global variable **ZIP** (on the right).
- When you run the application, the value of the **zip** attribute is assigned to the global variable **ZIP**.
- Click **OK** to close the window.



# Agenda

- Creating a project with a hybrid application
- Weather application overview
- Building the **Home** view
- Building the **Weather** view
- Building the **Forecast** view
- **Deploying and previewing**

## *Deploying and Previewing*

- Your application is ready!
- Save the file (Ctrl-S).
- Deploy the application:
  - In **WeatherProject > apps**, right-click **Weather**, and click **Run As > Run on Worklight Development Server**.
- Deploy the adapter:
  - In **WeatherProject > adapters**, right-click **SoapAdapter1**, and click **Run As > Deploy Worklight Adapter**.
- Preview the app:
  - In **WeatherProject > apps**, right-click **Weather**, and click **Run As > Preview**.

## Previewing the Application

- The first view (**Home**) of the application opens.
- Click **ZIP** to enter the ZIP code.
- A view containing a data entry field opens.
- Type, for example, 10007.
- Click **Home** in the application heading to get back to the **Home** view.





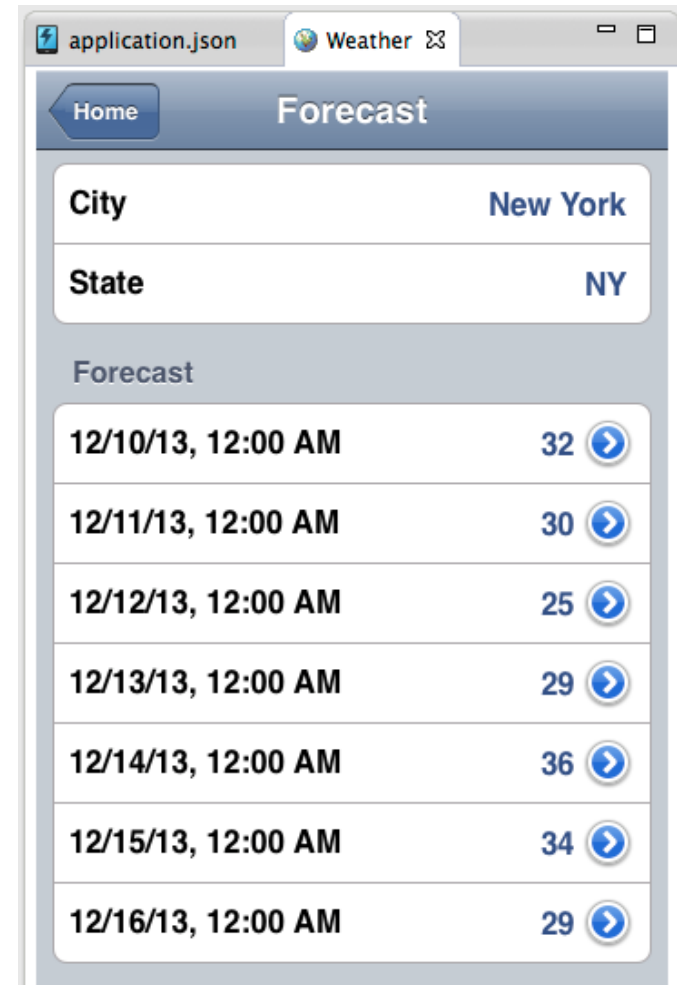
## Previewing the Application

- Click **Show Weather**.
- The second view (**Weather**) of the application opens.
- The data that is shown on this view is the result of the back-end service invocation.
- Click **Home** in the application heading to get back to the **Home** view.



## Previewing the Application

- Click **Show Forecast**.
- The third view (**Forecast**) of the application opens.
- The data that is shown on this view is the result of the back-end service invocation.
- To see more details about a specific date, click the corresponding row in the **Forecast** table.
- Click **Home** in the application heading to get back to the Home view.



## Wrap-up

- You created a simple hybrid mobile application.
- The application interacts with a public Web Service to get its data.
- The adapter that invokes the Web Service was automatically generated for your application.
- By using IBM Worklight Application Framework, you created the application data representation and the views without writing any code.
- Notes:
  - To run the application on a device, you must create an IBM Worklight environment for each platform. For more information about specific devices, see the modules *Previewing your application on <platform\_name>* under category 2, *Hello Worklight*, of the tutorials and samples.
  - If you import the sample project that is provided with this tutorial, make sure that you have **dojoLib** and **wlFrameworkLib** in your Eclipse workspace. These projects are created automatically when you create a hybrid application that uses IBM Worklight Application Framework.

## ***For more information***

- For more information about IBM Worklight Application Framework, see the section *Developing hybrid applications with IBM Worklight Application Framework* in:
  - [http://pic.dhe.ibm.com/infocenter/wrklight/v6r1m0/topic/com.ibm.worklight.dev.doc/dev/c\\_creating\\_hybrid\\_app\\_af.html](http://pic.dhe.ibm.com/infocenter/wrklight/v6r1m0/topic/com.ibm.worklight.dev.doc/dev/c_creating_hybrid_app_af.html)

# Notices

- Permission for the use of these publications is granted subject to these terms and conditions.
- This information was developed for products and services offered in the U.S.A.
- IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.
- IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
  - IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.
- For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:
  - Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan
- **The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.
- This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.
- Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
- IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.
- Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:
  - IBM Corporation  
Dept F6, Bldg 1  
294 Route 100  
Somers NY 10589-3216  
USA

- Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.
- The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.
- Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

## COPYRIGHT LICENSE:

- This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.
- Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:
  - © (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. \_enter the year or years\_. All rights reserved.

## Privacy Policy Considerations

- IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.
- Depending upon the configurations deployed, this Software Offering may use session cookies that collect session information (generated by the application server). These cookies contain no personally identifiable information and are required for session management. Additionally, persistent cookies may be randomly generated to recognize and manage anonymous users. These cookies also contain no personally identifiable information and are required.
- If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent. For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details> the sections entitled "Cookies, Web Beacons and Other Technologies" and the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

# Support and comments

- For the entire IBM Worklight documentation set, training material and online forums where you can post questions, see the IBM website at:
  - <http://www.ibm.com/mobile-docs>
- **Support**
  - Software Subscription and Support (also referred to as Software Maintenance) is included with licenses purchased through Passport Advantage and Passport Advantage Express. For additional information about the International Passport Advantage Agreement and the IBM International Passport Advantage Express Agreement, visit the Passport Advantage website at:
    - <http://www.ibm.com/software/passportadvantage>
  - If you have a Software Subscription and Support in effect, IBM provides you assistance for your routine, short duration installation and usage (how-to) questions, and code-related questions. For additional details, consult your IBM Software Support Handbook at:
    - <http://www.ibm.com/support/handbook>
- **Comments**
  - We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this document. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.
  - For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.
  - When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state.
  - Thank you for your support.
  - Submit your comments in the IBM Worklight Developer Edition support community at:
    - <https://www.ibm.com/developerworks/mobile/worklight/connect.html>
  - If you would like a response from IBM, please provide the following information:
    - Name
    - Address
    - Company or Organization
    - Phone No.
    - Email address

***Thank You***

