ptc university

Fundamentals of IoT Development with ThingWorx Exercise Workbook

M3:5. Model Exercise

1. Open ThingWorx Composer.
2. In the left pane, select the folder icon which opens the Browse menu.
3. In the MODELING section, cursor over **Things**. Notice the plus sign that appears to the right of Things.
4. Click the **plus sign** icon to create a new thing.
5. In the Name field, type **WeatherRover1**.

   Note: *Your username is automatically added at the end of any entity being created to avoid naming conflicts with the other users.*
6. In the Base Thing Template field, begin typing **GenericThing** and select **GenericThing** from the Search Results.
7. From the top menu, select the thing's **Properties and Alerts** tab.
8. Click **Add** to add your properties. On the right, a Property pane appears.
9. In the Name field, type **Battery**.
10. In the Base Type field, select **NUMBER**.
11. In the upper right of the pane, click  to finish the Battery property and then add a new property.
12. Add the following properties with the corresponding base types:

Name	Base Type
Speed	NUMBER
RawDestination	INTEGER
Destination	STRING

13. Click  to finish the Destination property.
14. Now that all the properties have been created, from the top menu, click **Save** to save the WeatherRover1 thing.

M3:8. Analyze Exercise

1. Verify that the **Recent** tab is selected. Notice the **WeatherRover1** thing under the Recent tab.
2. If necessary, click the **WeatherRover1** thing to open it.

   Note: *You can also open this thing using the Entity List area or using the SEARCH button at the top.*
3. In the WeatherRover1 thing near the top, select the **Subscriptions** tab.
4. Click the **Add** button to add your subscription.

5. In the Subscription Info tab in the left pane of the subscription editor:
   - Verify that **Me** is selected.
   - Select the **Enabled** check box.
6. In the left pane, select the **Inputs** tab of the subscription editor. In the Inputs tab:
   - In the Event field, type/select **DataChange**.
   - In the Property field, type/select **RawDestination**.
7. In the Subscription editing area, complete the following JavaScript code to update the Destination property for all input values of RawDestination. The following text contains the entire code snippet.

```
if(eventData.newValue.value == 0) {
  me.Destination = "On Base";
}
else if(eventData.newValue.value == 1) {
  me.Destination = "North Weather Station";
}
else if(eventData.newValue.value == 2) {
  me.Destination = "East Weather Station";
}
else if(eventData.newValue.value == 3) {
  me.Destination = "South Weather Station";
}
else if(eventData.newValue.value == 4) {
  me.Destination = "West Weather Station";
}
else if(eventData.newValue.value == 5) {
  me.Destination = "Dust Storm";
}
else {
  me.Destination = "unknown";
}
```

8. Click </> at the top of the editing area to check syntax and fix any issues.
9. After syntax passes successfully, click **Save and Continue** above the editing area.
10. Click **Done** above the editing area.
11. Click **Save** to save the updated WeatherRover1 thing.

M3:11. Connect Exercise

1. In the left, select the **Browse** tab and scroll down to the **SECURITY** section.
2. Cursor over **Application Keys** and click the **plus sign** icon to create a new application key.
3. In the Name field, type **ColonyKey**.
4. In the User Name Reference field, click the **plus** icon.

5. Select your **Username**.
6. Select a **date one week from now** in the Expiration Date field.
7. From the top, click **Save** to save the application key.
8. From the Windows start menu, select **ThingWorx Academic Simulator**.
9. Select Simulate Device.
10. Type your **Server URL** in the Server URL field.

    Note: *If you are using the Academic Cloud Manager instance, you can find your server URL by looking at the base URL of the ThingWorx server. Do not include "/Thingworx". For example: academic.cloud.thingworx.com. You may need to remove "http://" or "https://" from the beginning of the URL to connect.*
11. In ThingWorx, navigate back to the **ColonyKey**.
12. Copy the **Key ID** to the clipboard.
13. In the ThingWorx Academic Simulator window in the Application Key field, paste the **Key ID**.
14. In the Port field, type **80**.

    Note: *This depends on how the server has been configured to allow communication through. If SSL was enabled on your ThingWorx Server, then you would use port 443.*
15. Clear the Secured Server check box.

    Note: *If SSL is enabled on your ThingWorx Server, then select the* ***Secured Server*** *check box.*
16. Click Connect to ThingWorx.
17. Click the **green plus sign** to the right of Things (0).
18. In the Add Things From Platform field type/select **WeatherRover1**.

    Note: *This is the name of the Thing that we created in the model portion of this exercise. Make sure it matches the name exactly.*
19. Click **>>**. Notice that WeatherRover1 is populated in the Selected Things field.
20. Click Connect Things.
21. Click the **Speed** property row.
22. From the Simulation Type drop-down list, select **Step**.
23. In the Minimum Value field, type **0**.
24. In the Maximum Value field, type **20.**
25. In the Initial Value field, type **1.**
26. In the Step Size field, type **1**.

27. In the Refresh Rate (seconds) field, type **10**.
28. Select the **Include in Simulation** check box.
29. Click Save Property.
30. Create the following simulation data:

Property	Simulation Type	Minimum Value	Maximum Value	Initial Value	Step Size	Refresh Rate	Include in Simulation
Battery	Step	0	100	1	1	10	check
RawDestination	Step	0	5	0	1	10	check

Note: *Do not simulate Destination. This property is not sent by the edge device but is calculated by the subscription.*

31. At the top next to Properties, click the **play** button to simulate all properties.
32. Notice the values generated in the Value column. These values will refresh every ten seconds.
33. In ThingWorx, navigate to the WeatherRover1 thing.
34. Select the WeatherRover1 thing **Properties and Alerts** tab.
35. Click **Refresh**.
36. Wait 10 seconds and click **Refresh** again.
37. Notice the numbers in the Value column change and indicate that the data is being received from the "edge" (ThingWorx Academic Simulator) into your model. You have successfully connected your first device to ThingWorx!

M3:14. Build Exercise

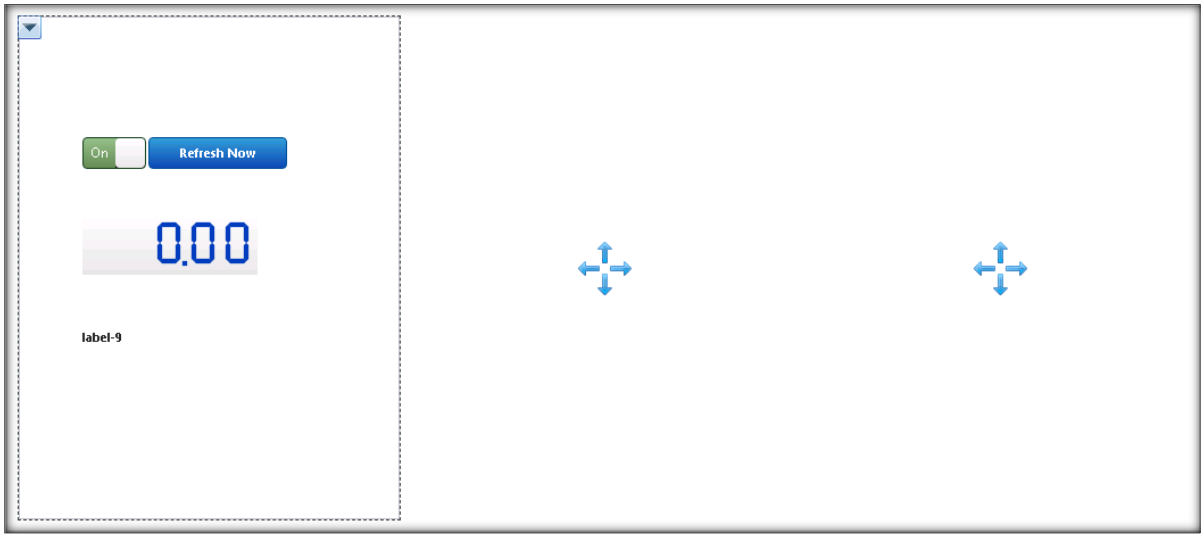
Create the mashup.

1. Click **+ NEW** at the top of the Composer. Type and select **Mashup** to create a new mashup.
2. Leave all defaults and click **OK**.
3. In the Name field, type **WeatherRoverMonitor**.
4. Click **Save**.

Design the layout.

5. Select the mashup's **Design** tab.
6. In the Filter Widgets field, type **lay**.

7. Drag-and-drop a **Layout** widget to the workspace.
8. From the Columns drop-down list, select **3**.
9. Click **Done**.
10. In the Filter Widgets field, clear lay, then type **auto**.
11. Drag-and-drop an **Auto Refresh** widget to the left of the three columns.
12. Click **Yes** to Add a Panel.
13. In the Filter Widgets field, clear the field.
14. Drag-and-drop an **LED Display** widget to the left column.
15. Drag-and-drop a **Label** widget to the left column below the LED Display widget.
16. Resize and reposition the widgets to look like this:



17. Drag-and-drop a **Gauge** widget to the middle column.
18. Drag-and-drop another **Gauge** widget to the right column.

Bind data to the widgets.

19. In the right side of the screen at the top in the Data tab, click the green plus sign to add an entity.
20. In the Select Entity field, begin typing **WeatherRover1** and select the **WeatherRover1** thing.
21. In the Select Services field, type **GetProperties**.
22. Click the **blue arrow** next to GetProperties to add the service.
23. Select the **Mashup Loaded?** check box.

24. Click **Done**.
25. In the right data sources pane, next to GetProperties, click the small **plus sign** to expand it.
26. From the properties, drag-and-drop **Speed** to the Gauge widget in the middle column.
27. A Select Binding Target drop-down menu will then appear to ask which "target" you want to bind the speed property to. Select **Data**.

    Note: *Notice in the Connections tab at the bottom of the screen, we see that Speed of the GetProperties service is bound to the Data attribute of the gauge.*
28. Drag-and-drop the **Battery** property to the Gauge widget in the right column.
29. Select **Data**.
30. Drag-and-drop the **RawDestination** property to the LED widget in the left column.
31. Select **Data**.
32. Drag-and-drop the **Destination** property to the Label widget in the left column.
33. Select **Text**.
34. Select the **Auto Refresh** widget.
35. Hover over the upper-left corner of the Auto Refresh widget to expand a drop-down list.
36. Drag the **Refresh** event to the GetProperties service in the Data tab.
37. If necessary, again select the **Auto Refresh** widget.
38. In the Widget Properties pane in the lower-left corner, in the RefreshInterval field delete 30, type **5**, then press **ENTER**.
39. Verify the RefreshInterval row is now green. This verifies that you have overridden the default value.
40. Scroll down and clear the Visible property check box.
41. Click **Save**.

View the mashup.

42. From the top, click **View Mashup**.

    Note: *It is likely that you will need to disable your popup blocker to view your finished mashup.*
43. In the upper-right corner of Chrome, click the popup blocker icon  .
44. Select **Always allow** pop-ups.
45. Click **Done**.
46. Click **View Mashup** again.

47. If the ThingWorx Academic Simulator is still running, you should see your values updating periodically. Congratulations, you just created your first IoT Application using ThingWorx!

M5:5. Create Base Rover Template

1. From the top menu, click **+NEW** to create a new Thing Template.
2. In the field, type/select **Thing Template**.
3. In the Name field, type **BaseRoverTemplate**.
4. In the Base Thing Template field, type/select **GenericThing**.
5. Click **Save**.
6. In the BaseRoverTemplate, select the Properties and Alerts tab.
7. Click **Add**.
8. Add the following properties with the corresponding base types:

Name	Base Type
Battery	NUMBER
Speed	NUMBER
RawDestination	INTEGER
Destination	STRING

9. Now that all the properties have been created, click **Save** at the top to save the BaseRoverTemplate template.

M5:6. Create Weather Rover

Create Weather Rover Template.

1. Click **NEW** on the Composer header and select **Thing Template** to create a new thing template.
2. In the Name field, type **WeatherRoverTemplate**.
3. In the Base Thing Template field, type/select **BaseRoverTemplate** which you previously created.
4. Select the **Properties and Alerts** tab. Notice that the properties in the BaseRoverTemplate are listed as Inherited Properties.
5. Click the **Add** button to add the following properties:

Name	Base Type
IsThreat	BOOLEAN
LastAlertLocation	STRING

6. Click **Save** at the top to save the WeatherRoverTemplate template.

Delete the WeatherRover1 Thing created earlier.

7. Select the **Browse** tab.
8. Under the MODELING section, select **Things**.
9. To the left of WeatherRover1, select the check box.
10. Click **Delete** at the top of the entities list.
11. Click **Yes** in the Delete the following entities? dialog box.

Create a WeatherRover1 thing based on the WeatherRoverTemplate.

12. Click **NEW** from the header of the Composer and type/select **Thing** to create a new thing.
13. In the Name field, type **WeatherRover1**.
14. In the Base Thing Template field, type/select **WeatherRoverTemplate**.
15. Click **Save**.
16. Select the thing's **Properties and Alerts** tab and notice the inherited properties from the WeatherRoverTemplate.

M5:7. Create Crew Rover

1. Create a new thing template with the name **CrewRoverTemplate**.
2. In the Base Thing Template field, type/select the **BaseRoverTemplate**.
3. Select the Properties and Alerts tab.
4. Click **Add**.
5. Type **MaxPassengers** in the Name field.
6. Type/select **INTEGER** for the Base Type field.
7. Select the **Has Default Value** check box and type **4** in the Has Default Value field.
8. Add the following property.

Name	Base Type	Has Default Value	Default Value
CurrentPassengers	INTEGER	checked	0

9. Click **Save** at the top to save the CrewRoverTemplate template.
10. Select the **Browse** tab.
11. Cursor over **Things** under the MODELING section. Notice the plus sign that appears to the right of Things.

12. Click the **plus sign** icon to create a new thing.
13. In the Name field, type **CrewRover1**.
14. In the Base Thing Template field, type/select **CrewRoverTemplate**.
15. Click **Save**.
16. On the right, click the **Browse** tab.
17. Select the **CrewRover1** thing check box.
18. Click **Duplicate** at the top of the entities list.
19. In the Name field, type **CrewRover2**.
20. Click **Save**.
21. Create another copy of CrewRover1 and give it the name **CrewRover3**.
22. Click **Save**.

M5:8. Create Weather Stations

1. Create a new thing template with the name **WeatherStationTemplate**.
2. Use the Base Thing Template GenericThing.
3. Add the following property with the corresponding base type:

Name	Base Type
WindSpeed	NUMBER

4. Click **Save** at the top to save the WeatherStationTemplate template.
5. Create the following four Thing entitiess using the WeatherStationTemplate:
   - NorthWeatherStation
   - SouthWeatherStation
   - EastWeatherStation
   - WestWeatherStation
6. Recall that after you create the NorthWeatherStation, you can duplicate it to create the others. Do not forget to save!

M5:9. Add Services to Crew Rover

1. Open the CrewRoverTemplate we created earlier.
2. Select the **Services** tab.

3. Click **Add**.
4. In the Name field, type **AddPassenger**.
5. On the left of the Services pane, select the **Output** tab.
6. Click the drop-down list in which NOTHING is selected and type/select **STRING**.
7. Type the following code in the editing area:

```
if(me.CurrentPassengers < me.MaxPassengers) {
    me.CurrentPassengers++;
    result = "Successfully added passenger.";
}
else {
    result = "Unable to add passenger. Rover is already full.";
}
```

8. Click </> at the top of the editing area to check syntax and fix any issues.
9. After syntax passes successfully, click **Save and Continue** above the editing area.
10. Click **Done**.
11. Click **Add**.
12. In the Name field, type **RemovePassenger**.
13. Select the **Output** tab.
14. Click the drop-down list in which NOTHING is selected and select **STRING** from the drop-down list.
15. Type the following code in the editing area:

```
if(me.CurrentPassengers > 0) {
    me.CurrentPassengers--;
    result = "Successfully removed passenger.";
}
else {
    result = "Unable to remove passenger. Rover is already empty.";
}
```

16. Click </> at the top of the editing area to check syntax and fix any issues.
17. After syntax passes successfully, click **Save and Continue** above the editing area.
18. Click **Done**.
19. Click **Save**.

M5:10. Test Crew Rover Services

1. Make sure you have saved the CrewRoverTemplate.
2. **Reload the browser** to allow your template changes to update in the CrewRover things.

3. Navigate back to one of your CrewRover things such as CrewRover1 from the **Recent** tab. It can be any of them since they are all inheriting from the same template.
4. Select **Services**.
5. Click the **play** ▶ icon in the Execute column for the AddPassenger service.
6. Click **Execute**.

   If all went correctly, you should be given a results box that shows the text “Successfully added passenger”. If not, then go back to check if there were any spelling or syntax errors in your service code.
7. To test the maximum passengers, click **Execute** four more times.

   On the fifth execution of the service you should get the following result: "Unable to add passenger, Rover is already full”.
8. Click **Done**.
9. Select the thing’s **Properties and Alerts** tab.

   You should see that the CurrentPassengers property value is 4. This is the result of executing the AddPassenger service until the rover was full.
10. Select **Services**.
11. Click the **play** ▶ icon in the Execute column for the RemovePassenger service.
12. Click **Execute**.

    If all went correctly, you should be given a results box that shows the text “Successfully removed passenger”. If not, then go back to check if there were any spelling or syntax errors in your service code.
13. To test the maximum passengers, click **Execute** four more times.

    On the fifth execution of the service you should get the following result: " Unable to remove passenger, Rover is already empty”.
14. Click **Done**.
15. Select the thing’s **Properties and Alerts** tab.

    You should see that the CurrentPassengers property value is 0.

M6:4. Analyze Wind Speed

Create a HighWindSpeed Alert for the WeatherStationTemplate.

1. Open the WeatherStationTemplate.
2. Select the Properties and Alerts tab.
3. Select the **Alerts** tab.
4. Click **Add**.
5. In the Name field, type **HighWindSpeed**.
6. From the Property drop-down list, select **WindSpeed**.
7. From the Alert Type drop-down list, select **Above**.
8. In the Limit field, type **75**.
9. Click ✓ to finish creating the alert.
10. Click **Save** to finish editing the template.

Set the LastAlertLocation for in the WeatherRoverTemplate.

11. Edit the WeatherRoverTemplate.
12. Select the **Subscriptions** tab.

Set the LastAlertLocation for the NorthWeatherStation when its HighWindSpeed alert is triggered.

13. Click **Add**.
14. In the Subscription Info tab, for Source, select the **Other entity** radio button.
15. In the Entity field, type/select **NorthWeatherStation**.
16. Select the **Enabled** check box.
17. Select the **Inputs** tab.
18. Click the **Event** drop-down list and select **Alert** from the list.
19. Click the **Property** drop-down list and type/select **WindSpeed** from the list.
20. Type the following in the editing area:

```
me.LastAlertLocation = "North Weather Station";
me.Destination = "Dust Storm";
```

21. Click **Done**.

Set the LastAlertLocation for the EastWeatherStation when its HighWindSpeed alert is triggered.

22. Edit the WeatherRoverTemplate.
23. Select the **Subscriptions** tab.
24. Click **Add**.
25. In the Subscription Info tab, for Source, select the **Other entity** radio button.
26. In the Entity field, type/select **EastWeatherStation**.
27. Select the **Enabled** check box.
28. Select the **Inputs** tab.
29. Click the **Event** drop-down list and select **Alert** from the list.
30. Click the **Property** drop-down list and type/select **WindSpeed** from the list.
31. Type the following in the editing area:

```
me.LastAlertLocation = "East Weather Station";
me.Destination = "Dust Storm";
```

32. Click **Done**.

Set the LastAlertLocation for the remaining stations when the HighWindSpeed alert is triggered.

33. Create two more subscriptions to the HighWindSpeed alert for the following locations:
    - SouthWeatherStation
    - WestWeatherStation
34. Click **Save** to finish editing the template.

Test the subscription.

35. Reload the browser window.
36. Open the NorthWeatherStation thing.
37. Select the Properties and Alerts tab.
38. In the WindSpeed property, under Value, click the ✎ icon to set the value.

39. In the Set value of property field, type **85** then click ✓

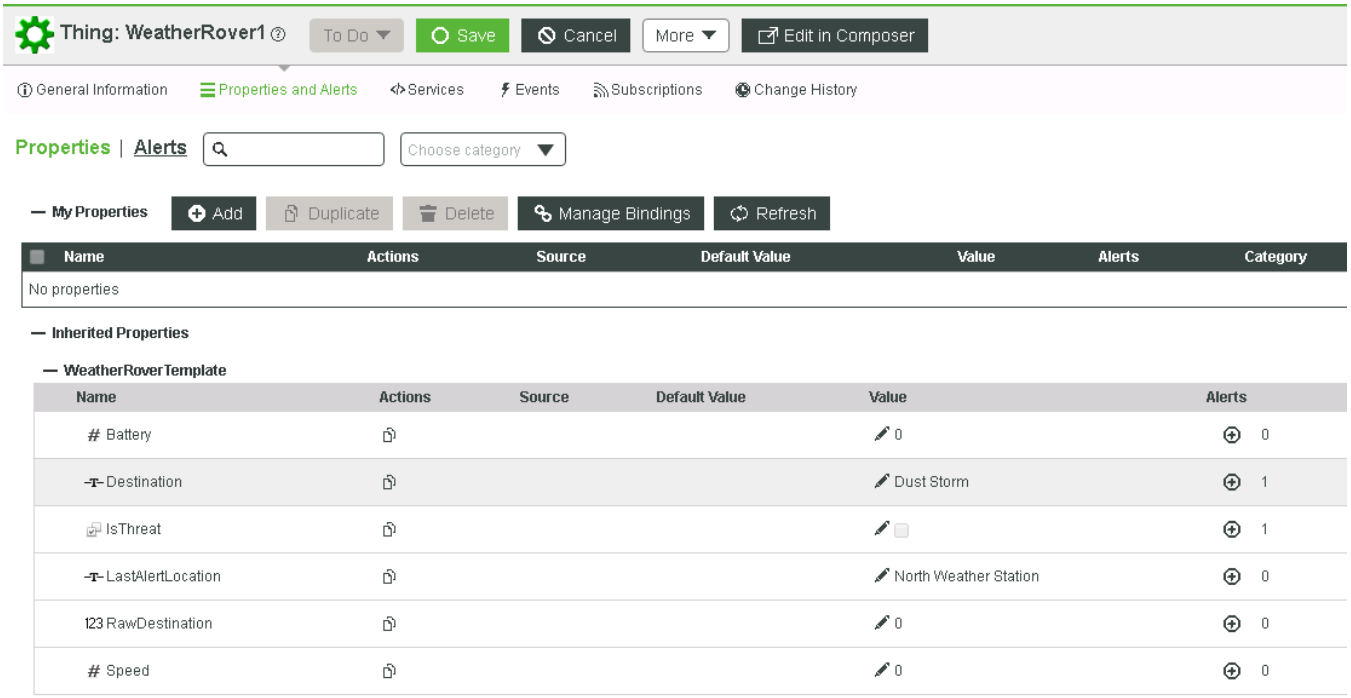
    You can see that the value has been updated to 85 and 1 Alert has been triggered.

40. In the left pane, select the clock icon that opens the Recent menu.
41. Hover over the **WeatherRover1** thing, to the right, click the X to close the cached entity.

    Note: *Whenever you update a template to a thing that is listed in the Recent pane, you must close it from the Recent pane to remove the cached thing. Then, open the thing from the Browse pane.*

42. From the Browse tab, open the **WeatherRover1** thing.
43. Select the Properties and Alerts tab.

    You should see that the Destination has changed to Dust Storm and the LastAlertLocation has changed to North Weather Station.



M6:5. Analyze Dust Storm Threat

1. Open the WeatherRoverTemplate.
2. Select the Properties and Alerts tab.
3. Select the **Alerts** tab.
4. Click **Add**.

5. Type **InStorm** in the Name field.
6. Select **Destination** from the Property drop-down list.
7. Select **EqualTo** from the Alert Type drop-down list.
8. Type **Dust Storm** in the Value field.
9. Click ✓ to finish creating the alert.
10. Select the **Subscriptions** tab.
11. Click **Add**.
12. Verify that the **Me** radio button is selected in the Subscription Info tab.
13. Select the **Enabled** check box.
14. Select the **Inputs** tab.
15. Click the **Event** drop-down list and select **Alert** from the list.
16. Click the **Property** drop-down list and select **Destination** from the list.
17. Type the following in the Script editing area:

```
me.IsThreat = Math.random() > 0.1;
```

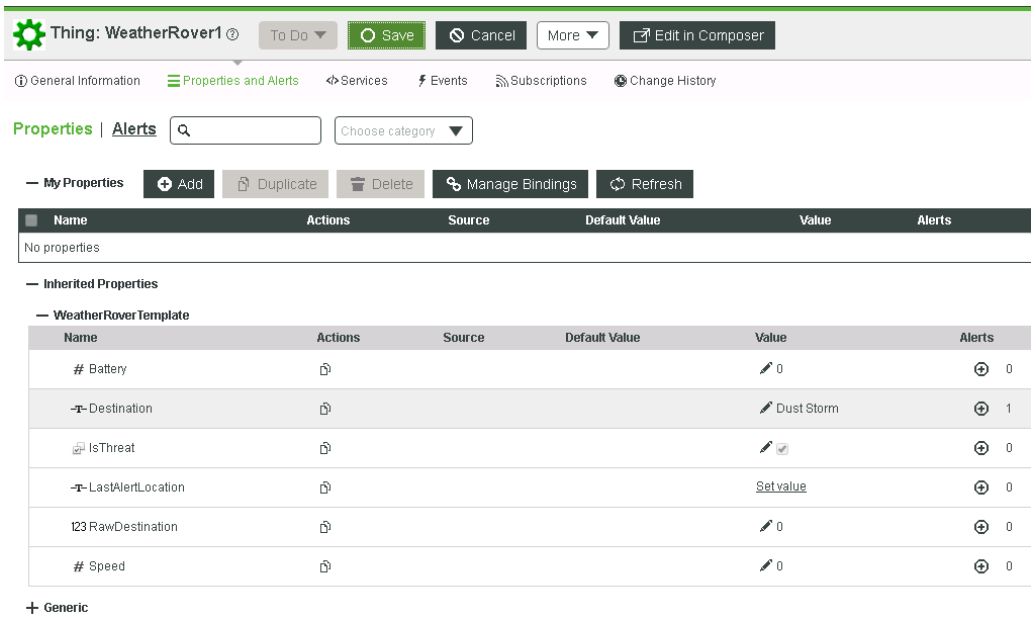
***Note**: This line of code is referencing the common JavaScript library, Math, and using its function, random. This function generates a random number between 0 and 1. So essentially, we are evaluating if a random number is greater than 1. This will turn out to be true 90% of the time. We chose 90% of the time simply because our application will be much more interesting if the dust storm is a threat!*

18. Click **Done**.
19. Click **Save** to finish editing the template.
20. Open the **WeatherRover1** thing.

***Note:** Whenever you update a template to a thing that is listed in the Recent pane, you must close it from the Recent pane to remove the cached thing. Then, open the thing from the Browse pane.*

21. Select the Properties and Alerts tab.
22. In the Destination property row, click **Set value**.
23. In the Set value of property field, type **Dust Storm**.
24. Click ✓
25. Click **Refresh**.
26. Click **Save**.

***Note**: You should see that the IsThreat value is set to True. If it did not change, change the value away from Dust Storm first and then set it to Dust Storm again.*



M6:6. Dispatch Crew Rover

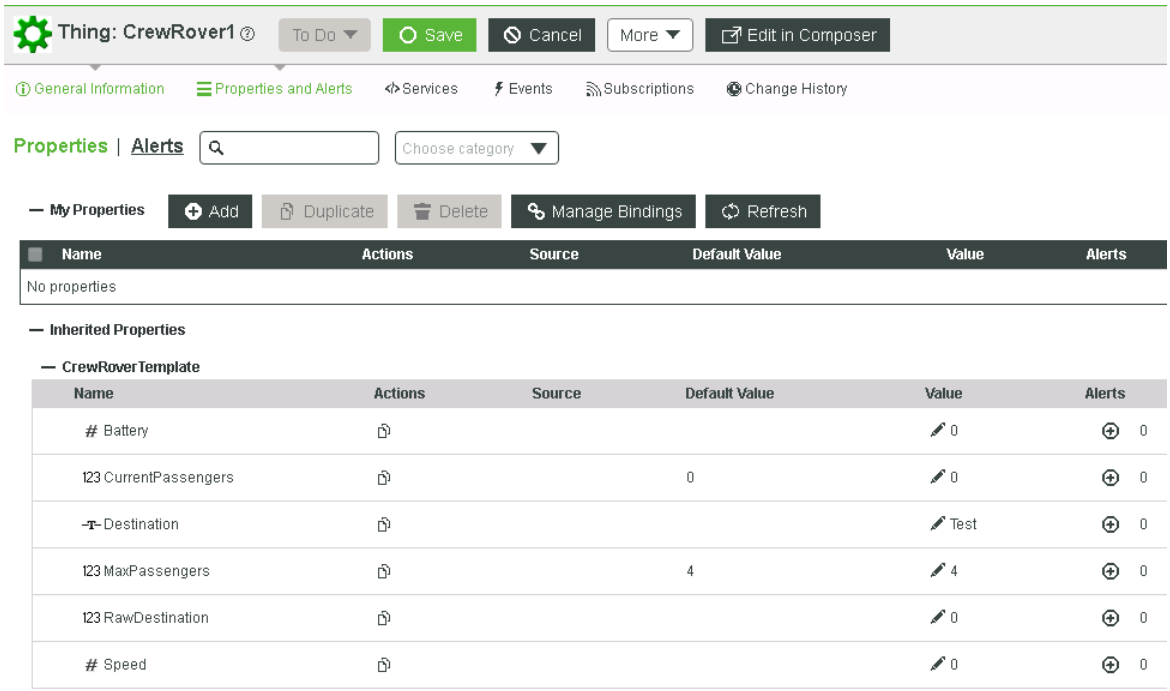
1. Edit the WeatherRoverTemplate.
2. Select the Properties and Alerts tab.
3. Select the **Alerts** tab.
4. Click **Add**.
5. In the Name field, type **DangerousStorm**.
6. From the Property drop-down list, select **IsThreat**.
7. From the Alert Type drop-down list, select **EqualTo**.
8. For the Value, select the **True** radio button.
9. Click ✓ to finish creating the alert.
10. Click **Save**.
11. Edit the CrewRoverTemplate.
12. Select the **Subscriptions** tab.
13. Click **Add**.
14. Select **Other entity** radio button for the Source field in Subscription Info tab.

15. In the Entity field, type/select **WeatherRover1**.
16. Select the **Enabled** check box.
17. Select the **Inputs** tab.
18. Click the Event drop-down list and select **Alert** from the list.
19. Click the Property drop-down list and select **IsThreat** from the list.
20. In the code below, replace *username* with your username. Type the following in the Script editing area:

```
me.Destination = Things["WeatherRover1_username"].LastAlertLocation;
```

21. Click **Done**.
22. Click **Save** to finish editing the template.
23. Refresh the web browser so that the alert and subscription are updated in the things.
24. Edit the **WeatherRover1** thing.
25. Select the Properties and Alerts tab.
26. In the LastAlertLocation property row, click **Set Value**.
27. In the Set value of property field, type **Test**.
28. Click ✓
29. Click **Save**.
30. Click **Refresh**.
31. In the IsThreat property row, click the pencil icon ✎.
32. If necessary, select the **False** radio button in the Set value of property field.
33. Click ✓
34. Click the pencil ✎ in the IsThreat property row again.
35. Select the **True** radio button in the Set value of property field.
36. Click ✓
37. Edit the **CrewRover1** thing.
38. Select the Properties and Alerts tab.
39. Click **Refresh**.

If the Destination property matches the value you typed above, Test, the CrewRover1 was deployed successfully.



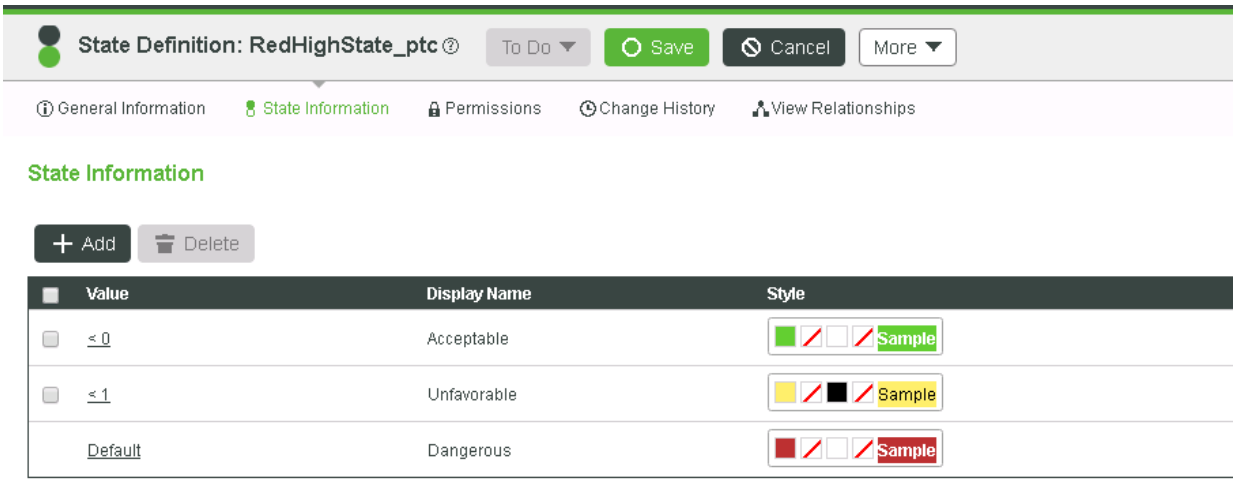
40. Edit the **CrewRover2** thing.
41. Select the Properties and Alerts tab.
42. Notice that the Destination property value is set to Test, this means that CrewRover2 was deployed successfully.
43. Verify that the CrewRover3 was deployed successfully.

M6:7. Giving Context to Our Data

1. Select the **Browse** tab.
2. Under the VISUALIZATION section, cursor over **State Definitions**. Notice the plus sign that appears to the right of State Definitions.
3. Click the **plus sign** icon to create a new state definition.

   ***Note**: You can also create a new state definition by clicking **NEW** in the Composer header and type/select **State Definition**.*
4. In the Name field, type **RedHighState**.
5. Click the State Information tab.
6. From the Apply State drop-down list, select **Numeric**.

7. Click **Add** under State Information.
8. In the Value field, type **0**.
9. In the Display Name field, type **Acceptable**.
10. In the Style field, type/select **noErrorStyleDefinition**.
11. Click ✓+ to finish creating the Acceptable state definition and add a new state definition.
12. In the Value field, type **1**.
13. In the Display Name field, type **Unfavorable**.
14. In the Style field, type/select **pendingStyleDefinition**.
15. Click ✓ to finish creating this state definition.
16. Click on **Default** from the state definition list under the State Information
17. In the Display Name field, type **Dangerous**.
18. In the Style field, type/select **errorStyleDefinition**.
19. Click ✓ to finish editing this state definition.
20. Click **Save** to finish the state definition.



21. Create a state definition named **GreenHighState** using the following parameters:

Value	Display Name	Style
< 0	Dangerous	errorStyleDefinition
< 1	Unfavorable	pendingStyleDefinition
Default	Acceptable	noErrorStyleDefinition

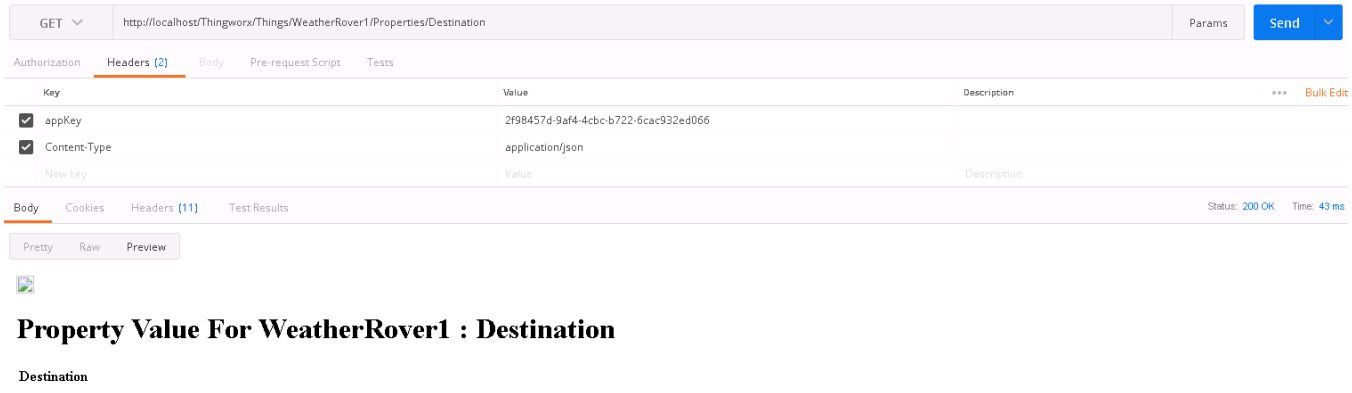
M7:7. Getting Started with REST

1. In Chrome, copy the ThingWorx URL up to and including **/Thingworx**.
2. Paste this URL into a document in a text editor, such as Notepad.
3. Select the **Browse** tab.
4. From the SECURITY section, select **Application Keys**.
5. Open ColonyKey.
6. If the selected date in the Expiration Date field is already expired, then select a **date one week from now**.
7. Click **Save**.
8. Copy and paste the **Key ID** of the ColonyKey into the text editor document.
9. Select the **Browse** tab.
10. From the MODELING section, select **Things**.
11. Copy and paste the following Thing names exactly as you entered them earlier to your text editor document.
    - CrewRover1_*username*
    - WeatherRover1_*username*
    - NorthWeatherStation_*username*
12. Select **Thing Templates** from the MODELING section.
13. Click WeatherStationTemplate.
14. Select the Properties and Alerts tab.
15. Select Properties.
16. Copy the **WindSpeed** property name exactly as you entered earlier to your text editor document.
17. From the left pane, click **CrewRoverTemplate**.
18. Select the Properties and Alerts tab.
19. Type the **CurrentPassengers** and **Destination** Property names exactly as you entered earlier to your text editor document.
20. Select the **Services** tab.
21. Copy the **AddPassenger** and **RemovePassenger** Service names exactly as you entered earlier to your text editor document.

M7:8. GET Request

1. Open **Postman** and close the pop-up window which appears.
2. Ensure that the http verb to the left of the Enter request URL field is set to **GET**.
3. In the Enter request URL field, type the URL that follows. Replace the bracketed text with the URL you copied to your text editor earlier. Replace *username* with your user name. Replace WeatherRover1 and/or Destination if your spelling is different.

   [copied URL]/Things/WeatherRover1_*username*/Properties/Destination
4. Select the **Headers** tab.
5. In the New key field, type **appKey**.
6. In the Value field, paste your Application Key (**Key ID**) from the ColonyKey.
7. On the next line, in the New key field, type/select **Content-Type**.
8. In the Value field, type/select **application/json**.
9. Click **Send**.
10. In the Body tab, click **Preview** to view the request.



M7:9 PUT Request

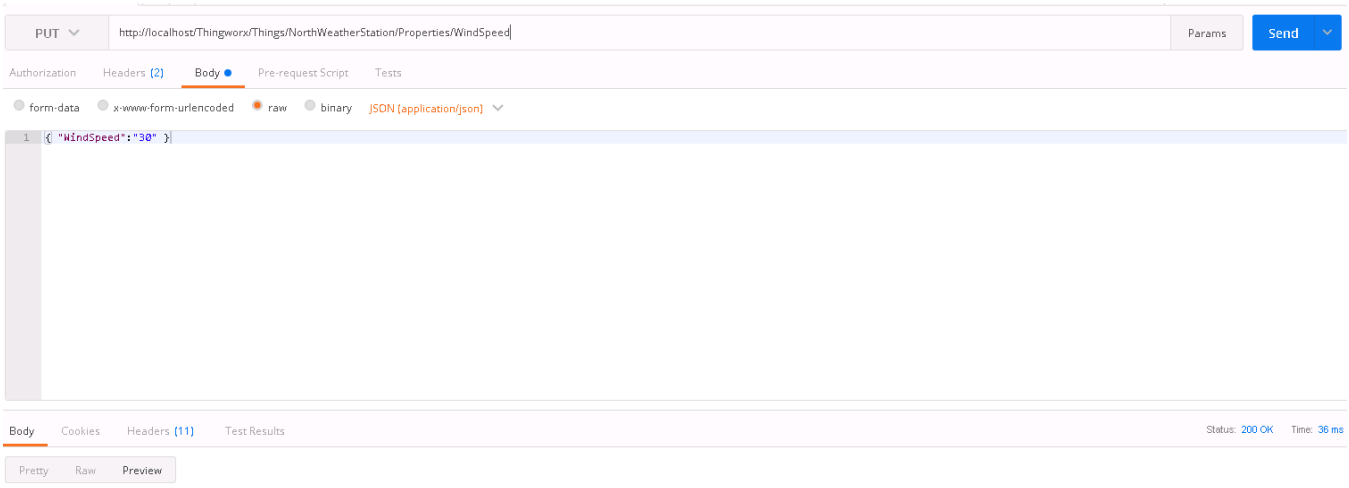
1. In Postman, click **GET** and select **PUT** from the drop-down list.
2. In the Enter request URL field, type the URL that follows. Replace the bracketed text with the URL you copied to your text editor earlier. Replace *username* with your user name. Replace NorthWeatherStation and/or Windspeed if your spelling is different.

   [copied URL]/Things/NorthWeatherStation_*username*/Properties/WindSpeed

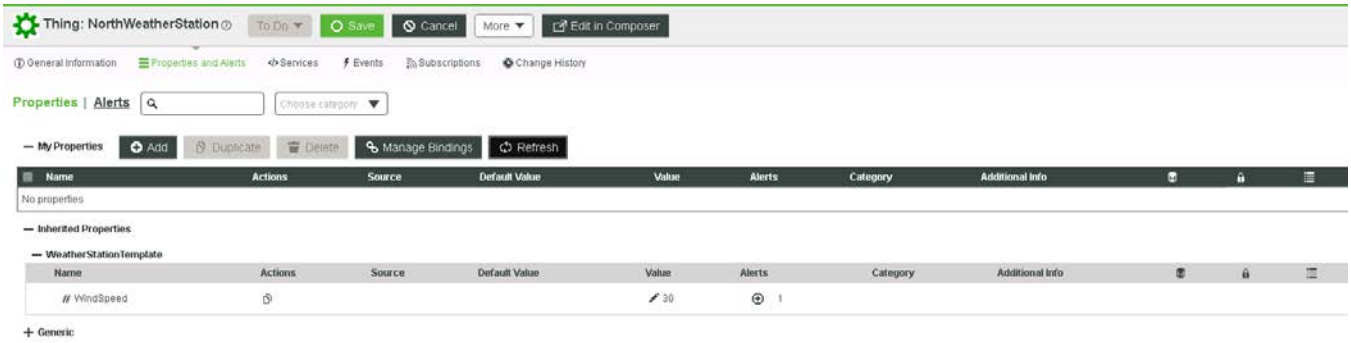
3. Select the **Body** tab to the right of the Headers tab.
4. Select the **raw** radio button.
5. Ensure that your formatting is JSON using the drop-down list on the right.
6. In the text editing area, type:

```
{ "WindSpeed":"30" }
```

7. Click **Send**.



8. Navigate back to the Composer.
9. View the NorthWeatherStation Thing.
10. Select the Properties and Alerts tab.
11. Click **Refresh**.
12. Notice the WindSpeed property Value is now 30.

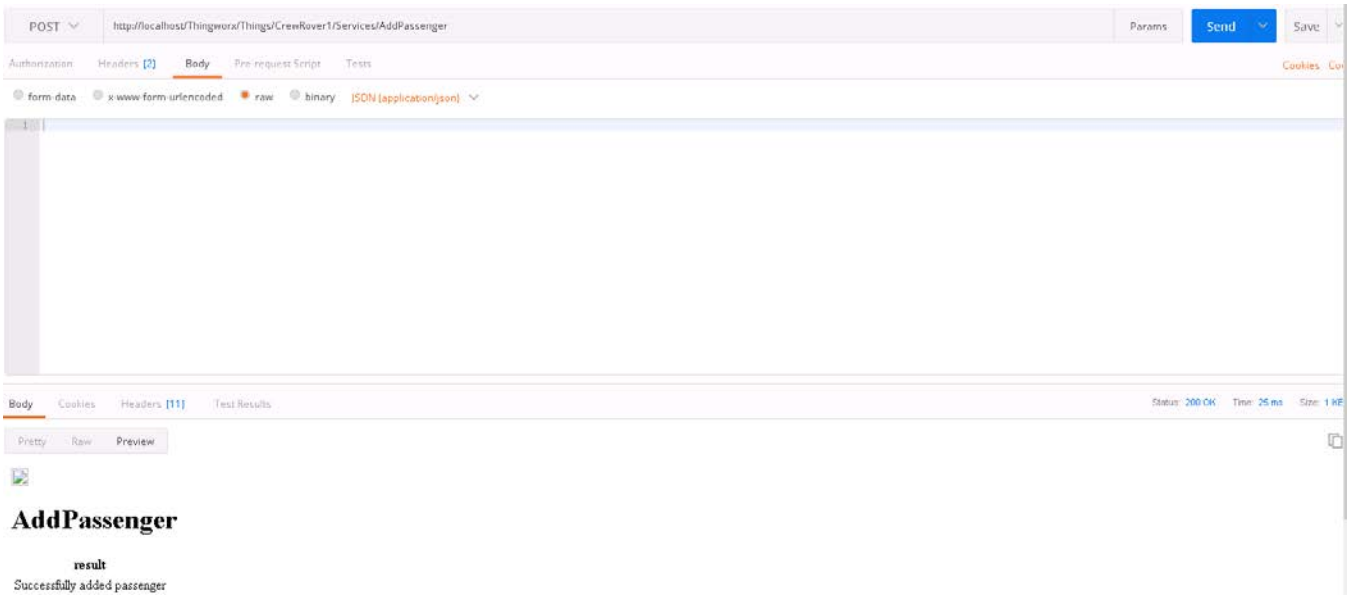


M7:10. POST Request

1. In Postman, click **PUT** and select **POST** from the drop-down list.

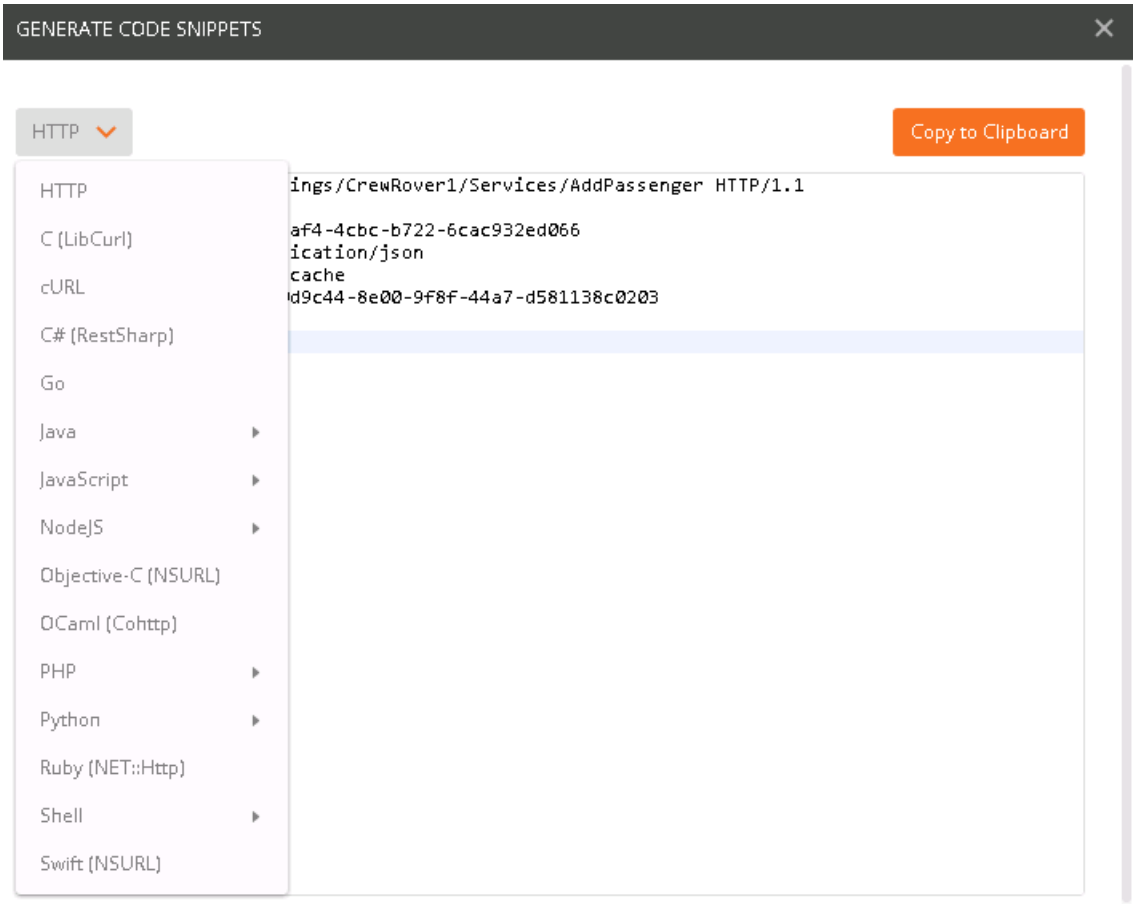
2. In the Body tab near the top of the screen, delete everything you typed earlier from the text editing area.
3. In the Enter request URL field, type the URL that follows. Replace the bracketed text with the URL you copied to your text editor earlier. Replace *username* with your user name. Replace CrewRover1 and/or AddPassenger if your spelling is different.

   [copied URL]/Things/CrewRover1_*username*/Services/AddPassenger
4. Click **Send**.
5. Select the **Preview** tab above the bottom panel.

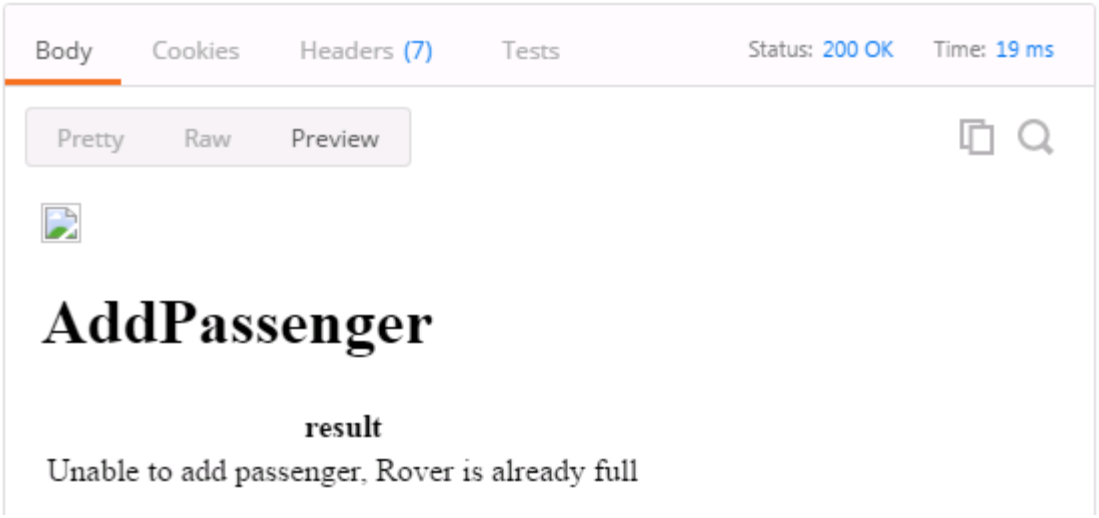


6. Click **Code** in the top right corner just near the Save button.
7. Click the drop-down list at the top and notice all the different programming languages. Select **HTTP**.

   This can be a useful tool for writing an edge application!



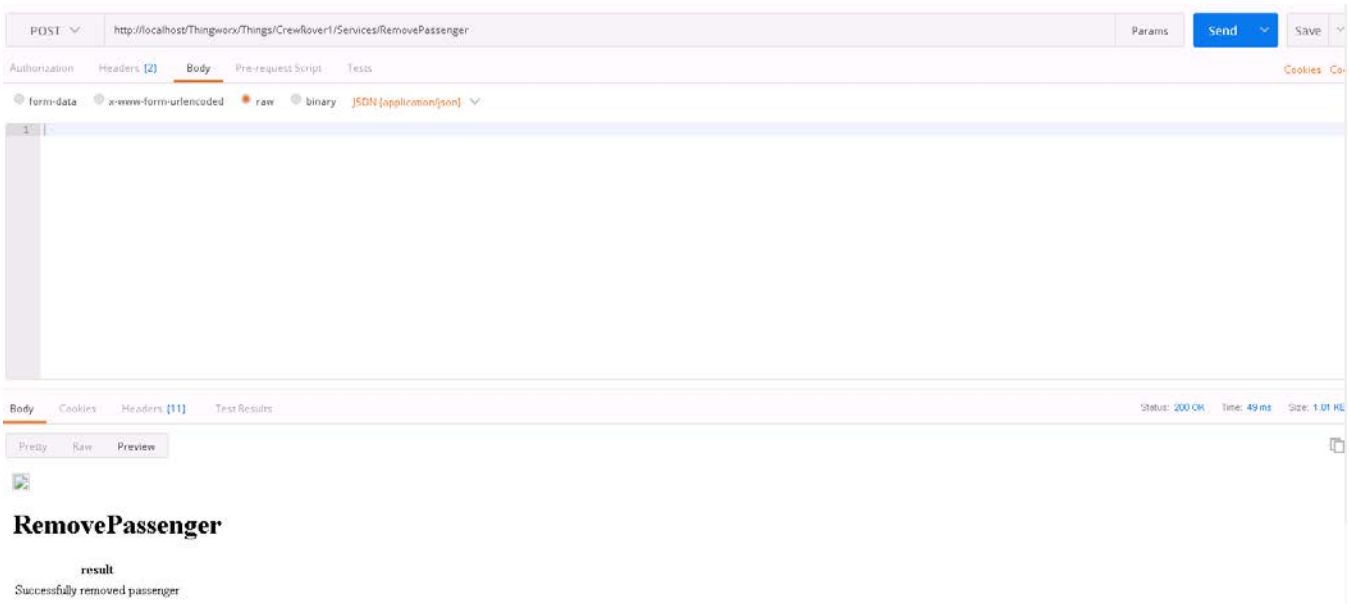
8. Close the Generate Code Snippets window.
9. Click **Send** four more times and notice the result in the Preview tab.



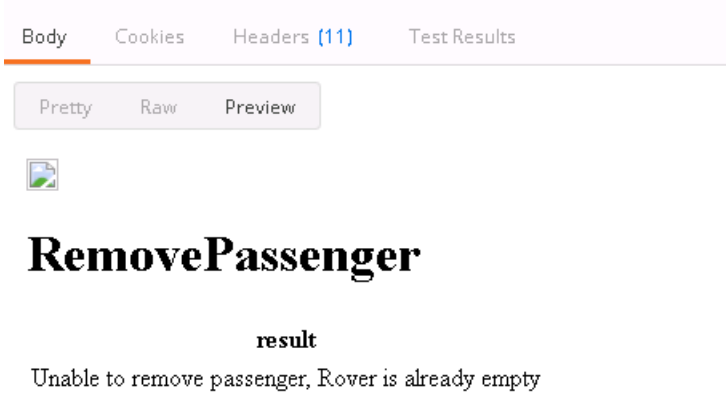
10. Replace the previous request URL with the following request URL:

    [copied URL]/Things/CrewRover1_*username*/Services/RemovePassenger

11. Click **Send**.



12. Click **Send** four more times and notice the result in the Preview tab.



M8:4. Major Mashup Components

Create a Mashup

1. Select the **Browse** tab.
2. Under the VISUALIZATION section, cursor over Mashups. Notice the plus sign icon that appears.
3. Click the **plus sign** icon to create a new mashup.
4. Leave all default values and click **OK**.
5. In the Name field, type **MarsColony**.
6. Click **Save**.
7. Select the **Design** tab.
8. In the Widget properties pane (located in the lower left corner of the mashup), locate the Master property.
9. In the Search Mashup field of the Master property, type/select **MarsMaster**.

   ***Note**: The Widget properties pane updates for the currently selected widget. The MarsMaster master mashup has been pre-built for this exercise.*
10. Click **Save** at the top of the screen to save all changes.
11. Click View Mashup.

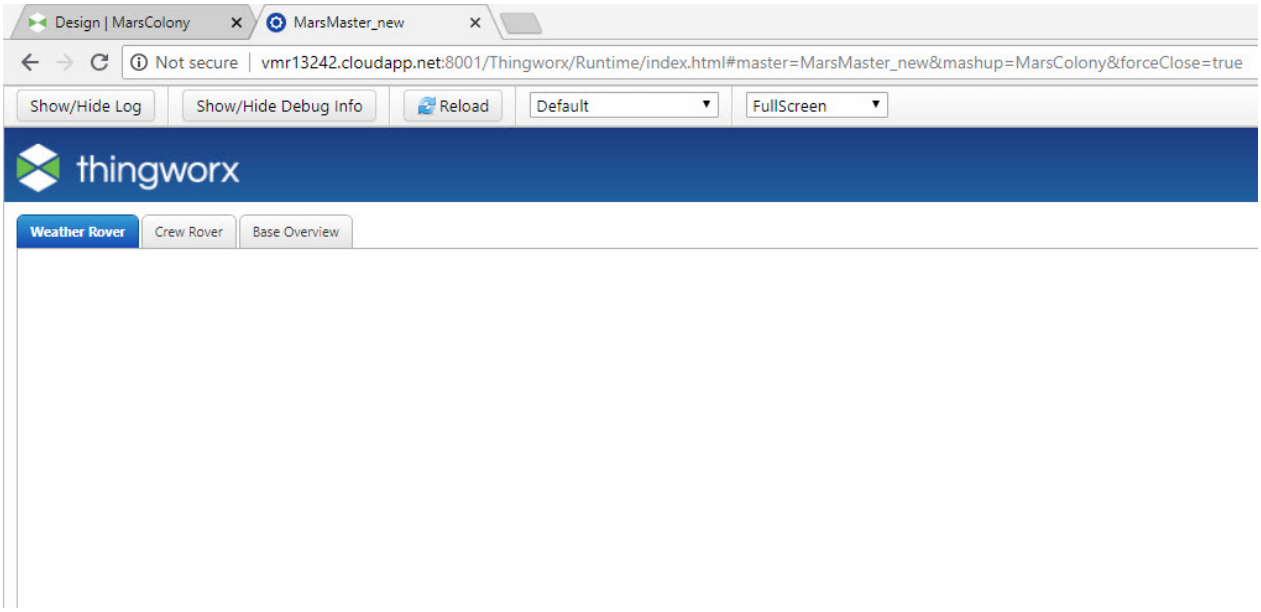
    Disable the pop-up blocker if necessary to view the mashup. Notice the banner at the top of the screen. This is inherited from the MarsMaster master mashup. A Master is a frame that contains a mashup which provides a way to consistently display items in different mashups, much like a template.
12. In Chrome, close the **MarsMaster** tab to close the mashup window.

Create Three Tabs

13. In the Filter Widgets field, type **Tabs**.
14. Drag-and-drop a **Tabs – Responsive** widget to the workspace.
15. In the Widget properties pane for the Tabs – Responsive widget, complete the following:
    - In the NumberOfTabs field, edit the value of 2 to **3** and then press TAB.
    - Scroll down the properties pane until the Tab Name fields are visible.
    - In the Tab1Name field, type **Weather Rover** and then press TAB.

- In the Tab2Name field, type **Crew Rover** and then press TAB.
- In the Tab3Name field, type **Base Overview** and then press TAB.

16. Click **Save**.
17. Click View Mashup.



18. In Chrome, close the **MarsMaster** tab to close the mashup window.
19. Verify that the mashup's Design tab is selected. If necessary, select the **Design** tab.
20. Select the **Weather Rover** tab.
21. In the Filter Widgets field, type **Panel**.
22. Drag-and-drop a **Panel** widget into the area directly below the Weather Rover tab.

    ***Note**: The four blue arrows that signify a responsive layout area have disappeared. This signifies you have properly placed the Panel widget.*

23. Select the **Crew Rover** tab.
24. Drag-and-drop a **Panel** widget into the area directly below the Crew Rover tab.
25. Select the **Base Overview** tab.
26. Drag-and-drop a **Panel** widget into the area directly below the Base Overview tab.

Add Images to the Backgrounds

27. Select the **Weather Rover** tab.

28. In the Filter Widgets field, type **Image**.
29. Drag-and-drop an **Image** widget on top of the panel.
30. Locate the SourceURL property in the Widget properties pane.
31. In the Search Media field, begin typing **WeatherRoverBackground** and then select the **WeatherRoverBackground** entity from the Search Results.

    ***Note**: The image appears in the image widget area, but it is too small.*

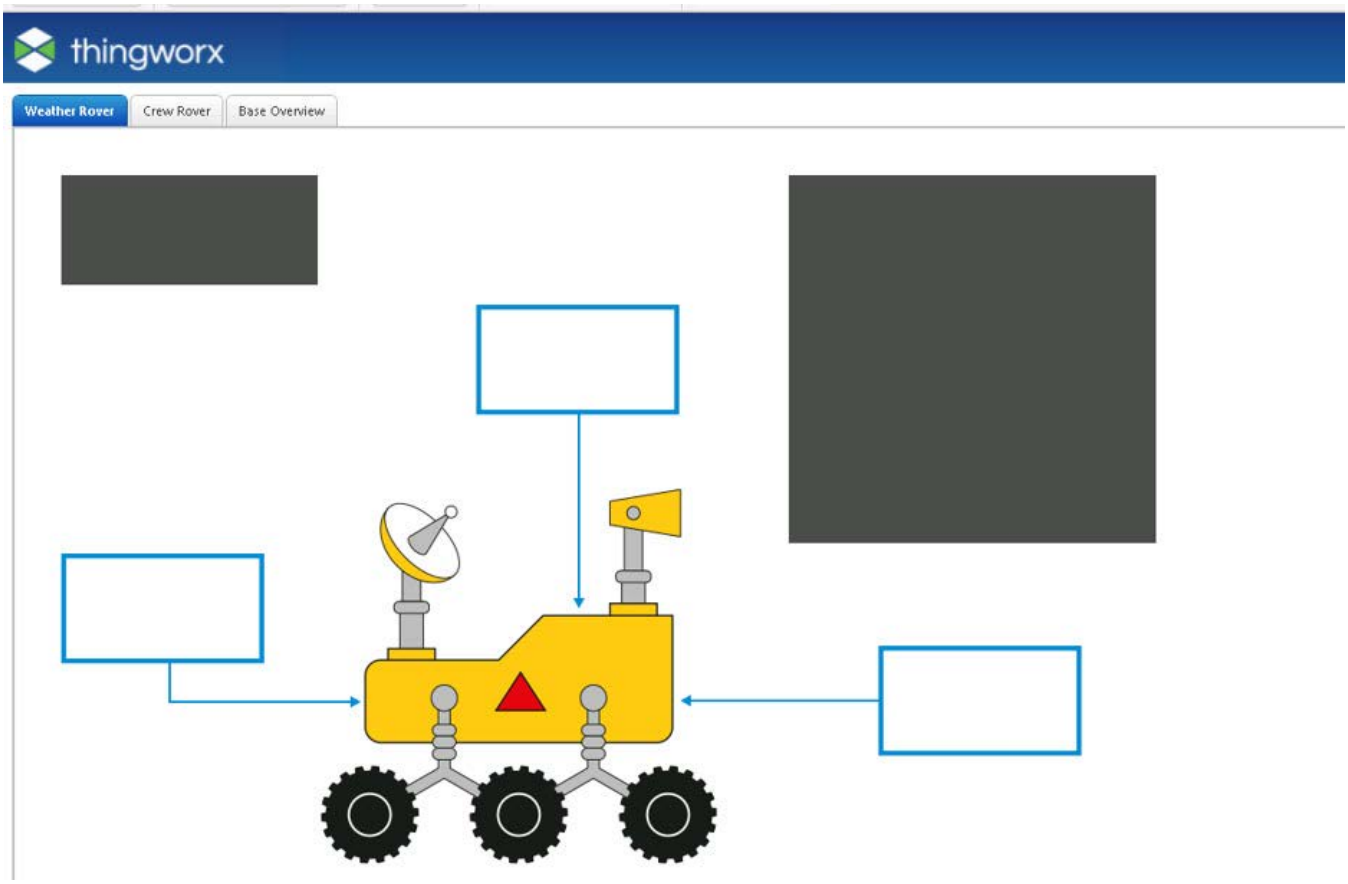
32. In the Widget properties pane for the Image widget, update the following property values. Remember after you type the value to press ENTER or TAB to update it.

Property	Value
Top	10
Left	10
Width	1000
Height	650
Z-index	1

    ***Note**: The Z-index property designates the priority placement of the widget. A higher number Z-index places the element further to the front of display. In this example, an index of 1 aims to keep the image behind the rest of the widgets to be placed.*

33. Click **Save**.

34. Click View Mashup.



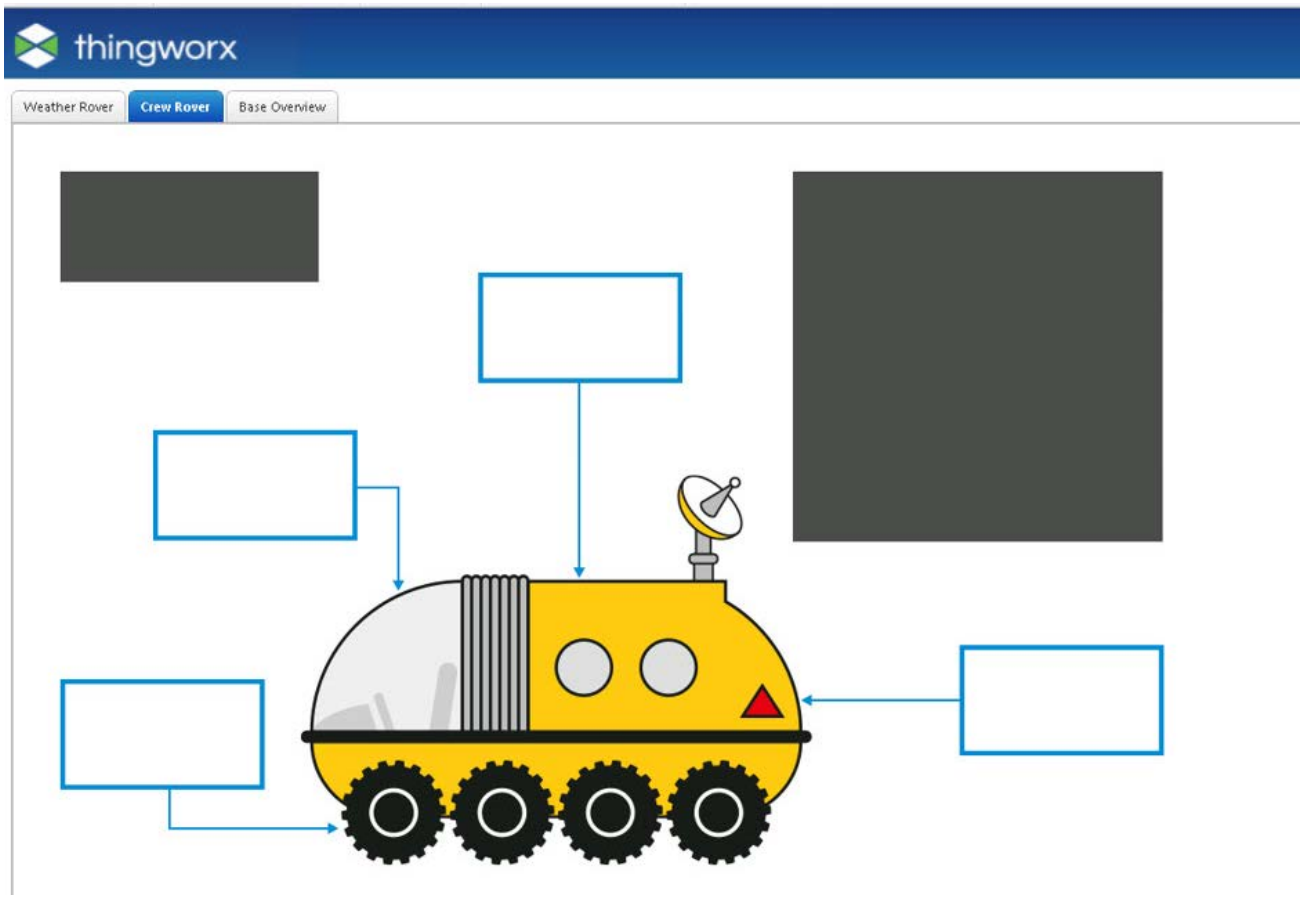
35. In Chrome, close the **MarsMaster** tab to close the mashup window.
36. In the left pane of the Mashup, select the **Workspace** tab.
37. Select the **image** widget from the Workspace tab.
38. With the image widget selected, click the **Copy** icon located in the upper-middle pane of the workspace.
39. Select the **Crew Rover** tab.
40. Select the **Panel** widget below the Crew Rover tab.
41. Click the **Paste** icon located in the toolbar above the workspace.

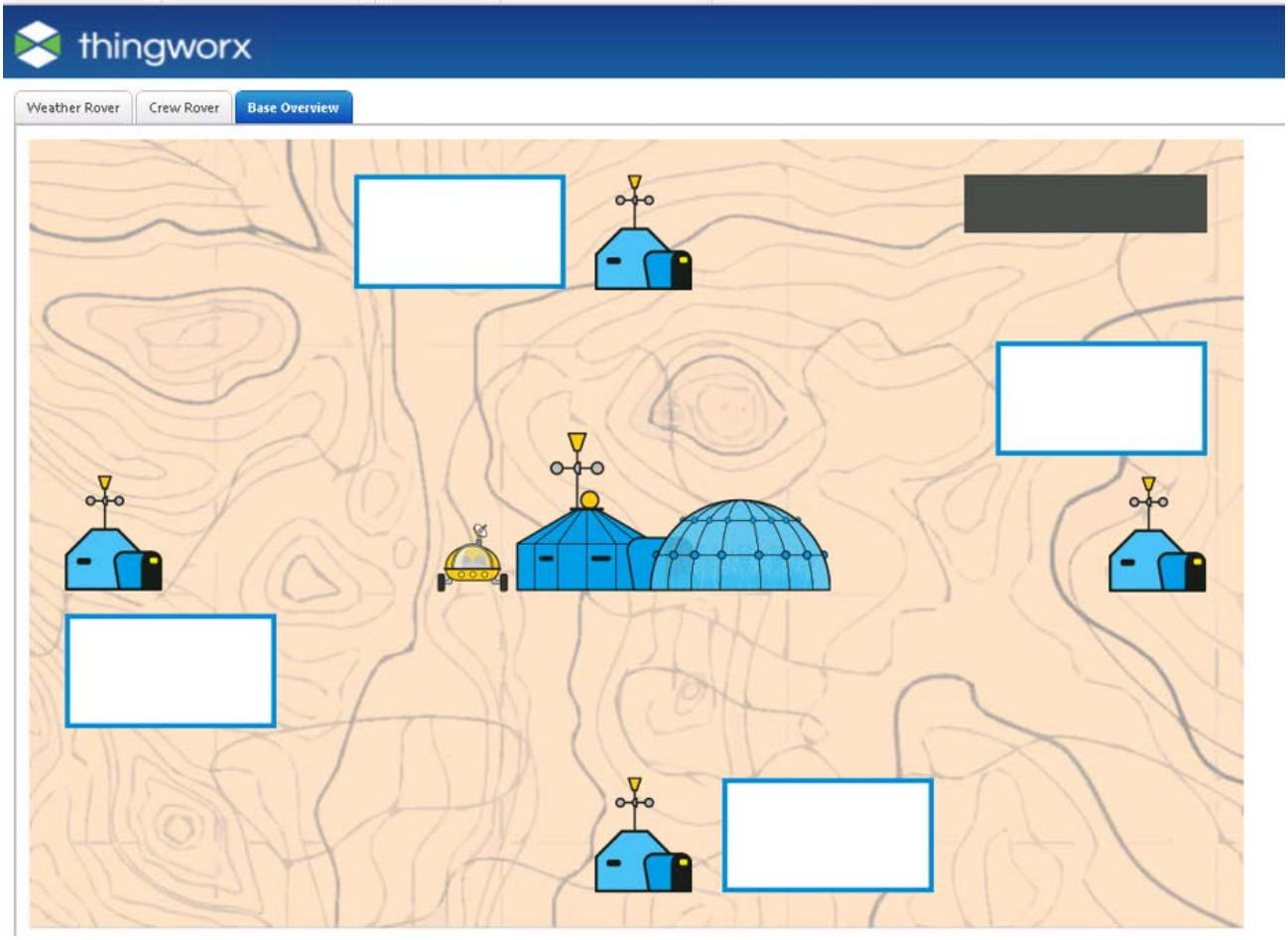
    ***Note**: The image widget is copied to the Crew Rover tab maintaining all the widget attributes set previously.*

42. In the Widget properties pane, locate the SourceURL property.
43. Click the **x** icon to clear the current SourceURL property.

44. In the Search Media field, begin typing **CrewRoverBackground** and then select the **CrewRoverBackground** entity from the Search Results.
45. Copy this image and paste it into the Base Overview panel.
46. In the Widget properties pane, change the SourceURL property to **BaseOverview**.
47. Click **Save**.
48. Click View Mashup.

    Navigate between the tabs and notice the blank sections in the overlay ready to be populated with widgets.





49. In Chrome, close the **MarsMaster** tab to close the mashup window.

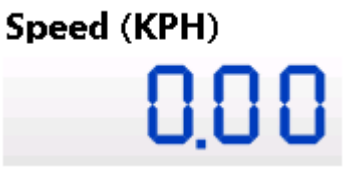
M8:5. Weather Rover Widget Setup

1. If the MarsColony mashup is not already opened, then open the MarsColony mashup for editing.
2. In the toolbar above the workspace, from the Preview drop-down list, select **1366x768**.
3. Select the **Weather Rover** tab.
4. In the left pane of the Mashup, select the **Widgets** tab.
5. Drag-and-drop an **LED Display** widget into the area next to the image widget.

   ***Note**: We cannot initially drop a widget on top of the image widget. First, we will place a widget into the workspace then position it in front of the image widget.*
6. Position the LED Display widget in the empty lower-left white box.

7. In the Widget properties pane for the selected LED Display widget, update the following properties:

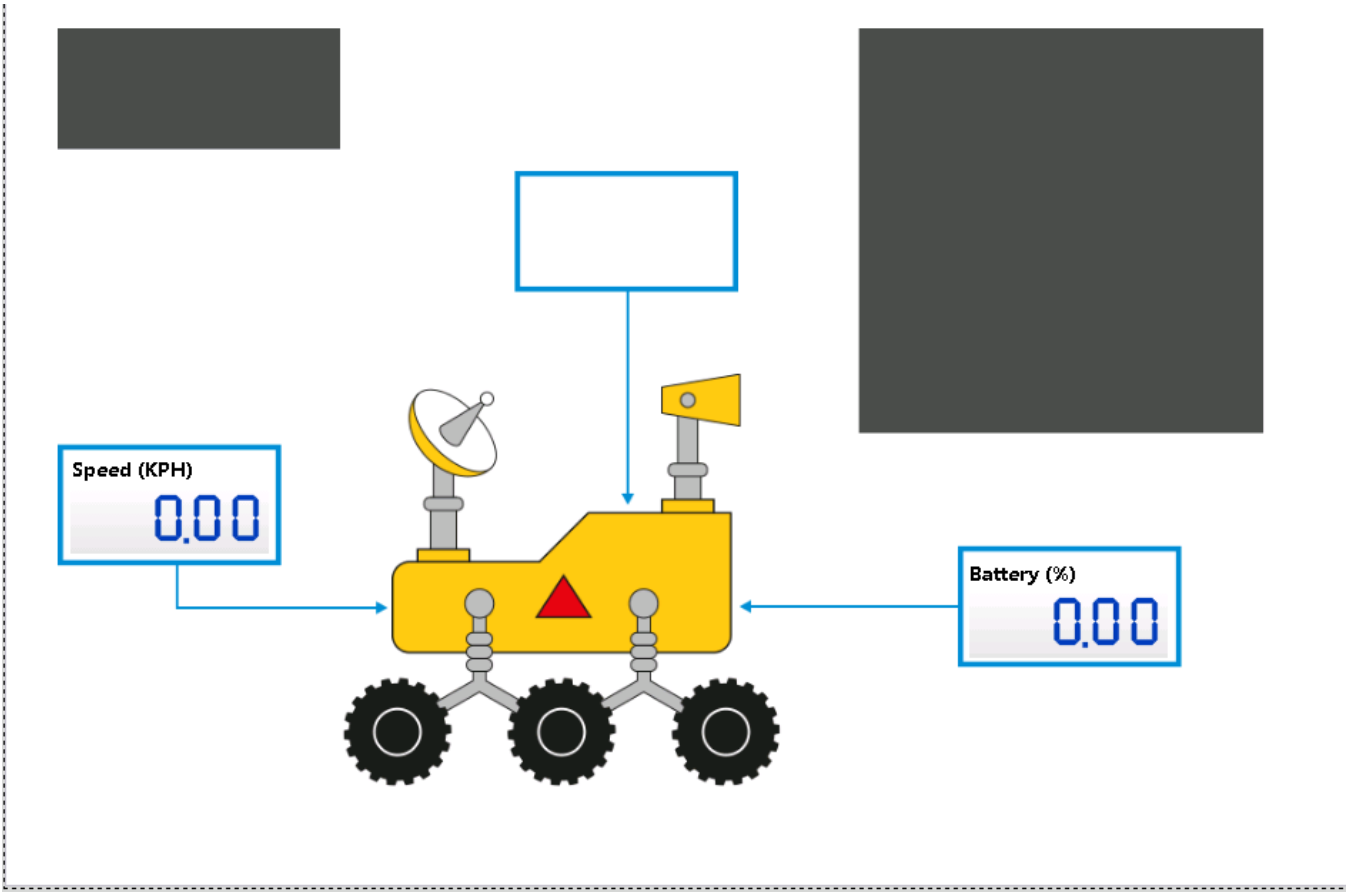
Property	Value
Label	Speed (KPH)
LEDLabelStyle	MarsMashupLabelStyle



8. Verify that the LED properties were updated.
9. Drag-and-drop another **LED** Display widget into the area next to the image widget.
10. Position the LED Display widget in the empty lower-right white box.
11. In the Widget properties pane for the selected LED Display widget, update the following properties:

Property	Value
Label	Battery (%)
LEDLabelStyle	MarsMashupLabelStyle

12. Drag-and-drop an **Auto Refresh** widget from the Widgets pane into the area next to the image widget.



13. Position the Auto Refresh widget near the bottom of the larger grey box.
14. From the Widgets pane, drag-and-drop a **Label** widget into the area next to the image widget.
15. Position the Label widget near the top of the larger grey box.

16. In the Widget properties pane for the selected Label widget, update the following properties:

Property	Value
Text	Travel Destination
Style	WhiteMarsMashupLabelStyle
Width	150

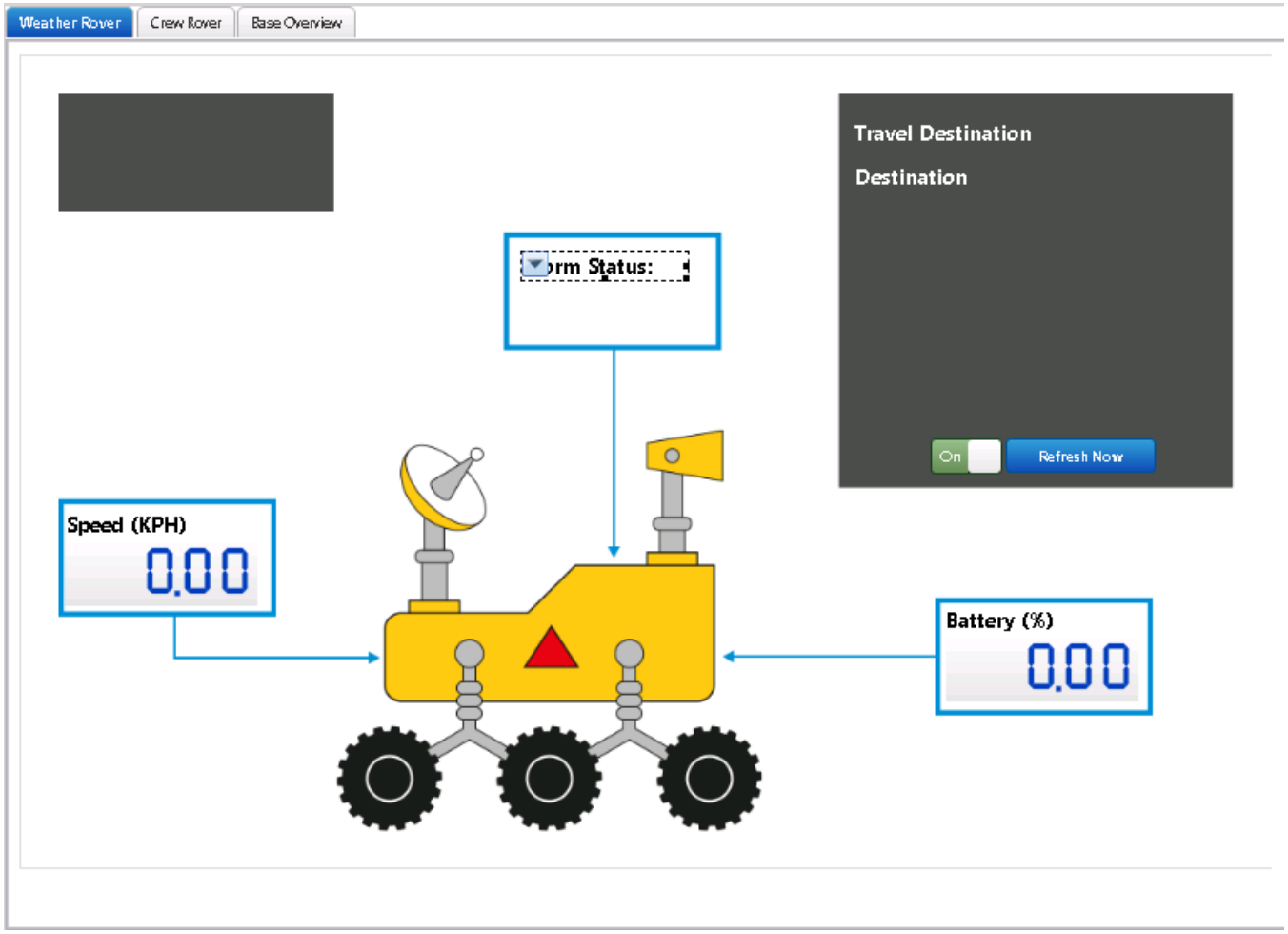
17. From the Widgets pane, drag-and-drop another **Label** widget into the area next to the image widget.
18. Position the Label widget below the Travel Destination Label widget.
19. In the Widget properties pane for the selected Label widget, update the following properties:

Property	Value
Text	Destination
Style	WhiteMarsMashupLabelStyle

20. Drag-and-drop another **Label** widget from the Widgets pane into the area next to the image widget.
21. Position the Label widget in the blank box pointing down to the Rover.
22. In the Widget properties pane of the label widget, update the following properties:

Property	Value
Text	Storm Status:
Style	MarsMashupLabelStyle

***Note**: Resize or reposition any of the label widgets to fit appropriately.*



23. From the Widgets pane, drag-and-drop a **List** widget into the area next to the image widget.
24. Position the List widget in the top-left grey box.
25. In the Widget properties pane for the selected List widget, update the following properties:

Property	Value
View	Dropdown
ListLabelStyle	WhiteMarsMashupLabelStyle
Label	Weather Rovers

26. From the Widgets pane, drag-and-drop a **Shape** widget into the area next to the image widget.
27. Position the Shape widget next to the Storm Status: label widget.

28. In the Widget properties pane for the selected Shape widget, update the following property:

Property	Value
ShapeType	Circle

29. Drag-and-drop an **Image** widget from the Widgets pane into the workspace.
30. Position the Image widget between the gray boxes.
31. In the Widget properties pane of the selected Image widget, update the following properties:

Property	Value
SourceURL	CriticalStormWarning
Left	272
Width	380
Height	70

32. Drag-and-drop a **Textbox** widget into the workspace.
33. Position the Textbox widget in the larger grey box below the second label.
34. In the Widget properties pane of the selected Textbox widget, update the following properties:

Property	Value
TextboxLabelStyle	WhiteMarsMashupLabelStyle
Label	Send to:

35. From the Widgets pane, drag-and-drop a **Checkbox** widget into the workspace.
36. Position the Checkbox widget in the larger grey box below the most recent widget.
37. In the Widget properties pane of the selected Checkbox widget, update the following properties:

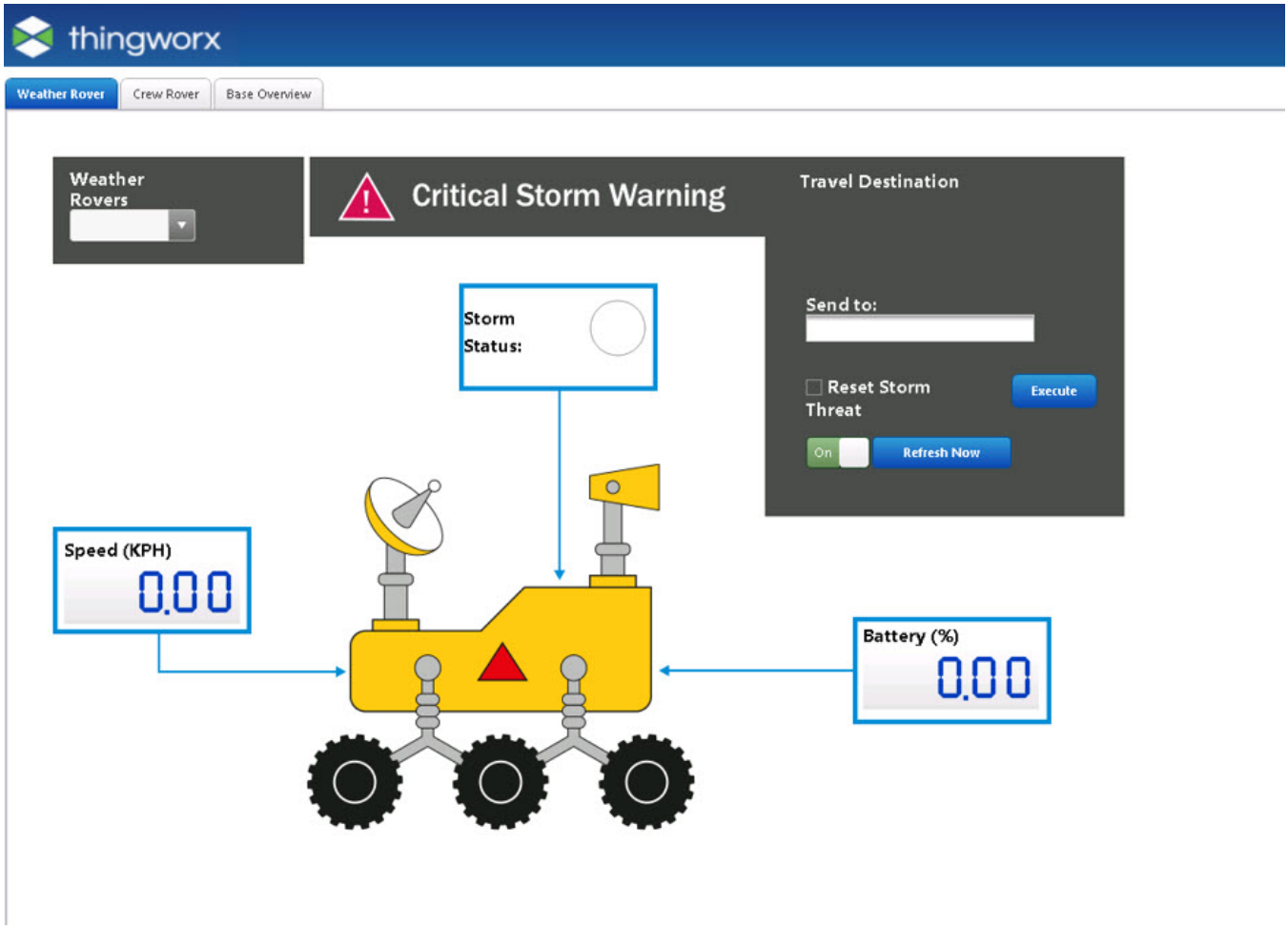
Property	Value
Prompt	Reset Storm Threat
Style	WhiteMarsMashupLabelStyle

38. Drag-and-drop a **Button** widget into the workspace.
39. Position the Button widget in the larger grey box under the checkbox widget.
40. In the Widget properties pane for the selected Button widget, update the following property:

Property	Value
Label	Execute

41. Click **Save**.

42. Click View Mashup.



43. Close the **MarsMaster** tab to close the mashup view window.

M8:6. Weather Rover Widget Binding

1. If the MarsColony mashup is not already opened, then open it.
2. If necessary, click **Edit** to edit the MarsColony mashup.
3. If necessary, select the **Design** tab.
4. In the top-right Entities pane, verify that the Data tab is selected, click the **green plus sign** icon to add an entity. An Add Data window appears.

   ***Note**: Data is brought into the mashup from entities to bind to the widgets.*

5. In the Select Entity field, type/select **WeatherRoverTemplate**.
6. In Select Services filter field, start typing **GetImplementingThingsWithData** and click the **blue arrow** next to it.

7. Select the **Mashup Loaded?** check box.
8. Click **Done**.
9. In the Data pane, verify that the GetImplementingThingsWithData service is already expanded. You should see a small minus sign to the left of GetImplementingThingsWithData. If the GetImplementingThingsWithData service is not already expanded, then click the **small plus sign** to the left of GetImplementingThingsWithData to expand it.
10. In the ThingTemplates_WeatherRoverTemplate > GetImplementingThingsWithData > Returned Data section, drag the **All Data** row to the Weather Rovers List widget.
11. Select **Data** as the Binding Target.
12. Select the **Weather Rovers** List widget.
13. In the Widget properties pane of the selected List widget, update the following properties:

Property	Value
AutoSelectFirstRow	selected
DisplayField	name
ValueField	name

14. In the upper right Data pane, click the **small plus sign** to expand the ThingTemplates_WeatherRoverTemplate > GetImplementingThingsWithData > Returned Data > Selected Row(s) section and drag the property to the widget using the binding target listed below:

Property	Widget	Binding Target
Speed	Speed LED Display	Data
Battery	Battery LED Display	Data
Destination	Destination Label	Text
IsThreat	Storm Status: Shape	Data
IsThreat	Reset Storm Threat Checkbox	State
IsThreat	Critical Storm Warning Image	Visible

***Note**: We just used the same property to bind three different kinds of widgets. When the Rover has a value of false for **IsThreat** or there is currently no storm that is threatening, the image will not be visible, and the checkbox will not be filled.*

15. Select the **Storm Status**: Shape widget.
16. In the Widget properties pane of the selected Shape widget, complete the following:
    - For the ShapeFormatter property, click **State Formatting**.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **IsThreat**.

- In the State Definitions Search field, type/select **ShapeThreatState**.
- Click **Done**.

***Note**: There are already pre-built state definitions which you can assign to widgets. You set up the shape widget so that when "IsThreat" is false, the shape is green and when "IsThreat" is true, the shape is red.*

17. Select the **Speed** LED Display widget.
18. In the Widget properties pane of the selected LED widget, complete the following:
    - For the LEDFormatter property, click **State Formatting**.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **Speed**.
    - In the State Definitions Search field, type/select **RedHighState**.
    - In the Value column, update the following values:

State	Value
Acceptable	< 10
Unfavorable	< 15

    - Click **Done**.
19. Select the **Battery** LED Display widget.
20. In the Widget properties pane of the selected LED widget, complete the following:
    - For the LEDFormatter property, click **State Formatting**.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **Battery**.
    - In the State Definitions Search field, type/select **GreenHighState**.
    - In the Value column, update the following values:

State	Value
Dangerous	< 10
Unfavorable	< 20

    - Click **Done**.
21. Select the **Auto Refresh** widget.
22. In the Widget properties pane for the selected Auto Refresh widget, update the following property:

Property	Value
RefreshInterval	5

23. Verify that the ThingTemplates_WeatherRoverTemplate > GetImplementingThingsWithData is expanded. Cursor over the arrow on the top-left corner of the Auto Refresh widget and drag the **Refresh** event to the GetImplementingThingsWithData service.
24. On the top right of the Entities pane, click the green **plus sign** icon. An Add Data window appears.
25. Select the **Dynamic** check box.
26. In the Select Entity field, type/select **WeatherRoverTemplate**.
27. In Select Services Filter field, begin typing **SetProperties** and click the **blue arrow** next to it.
28. Click **Done**.

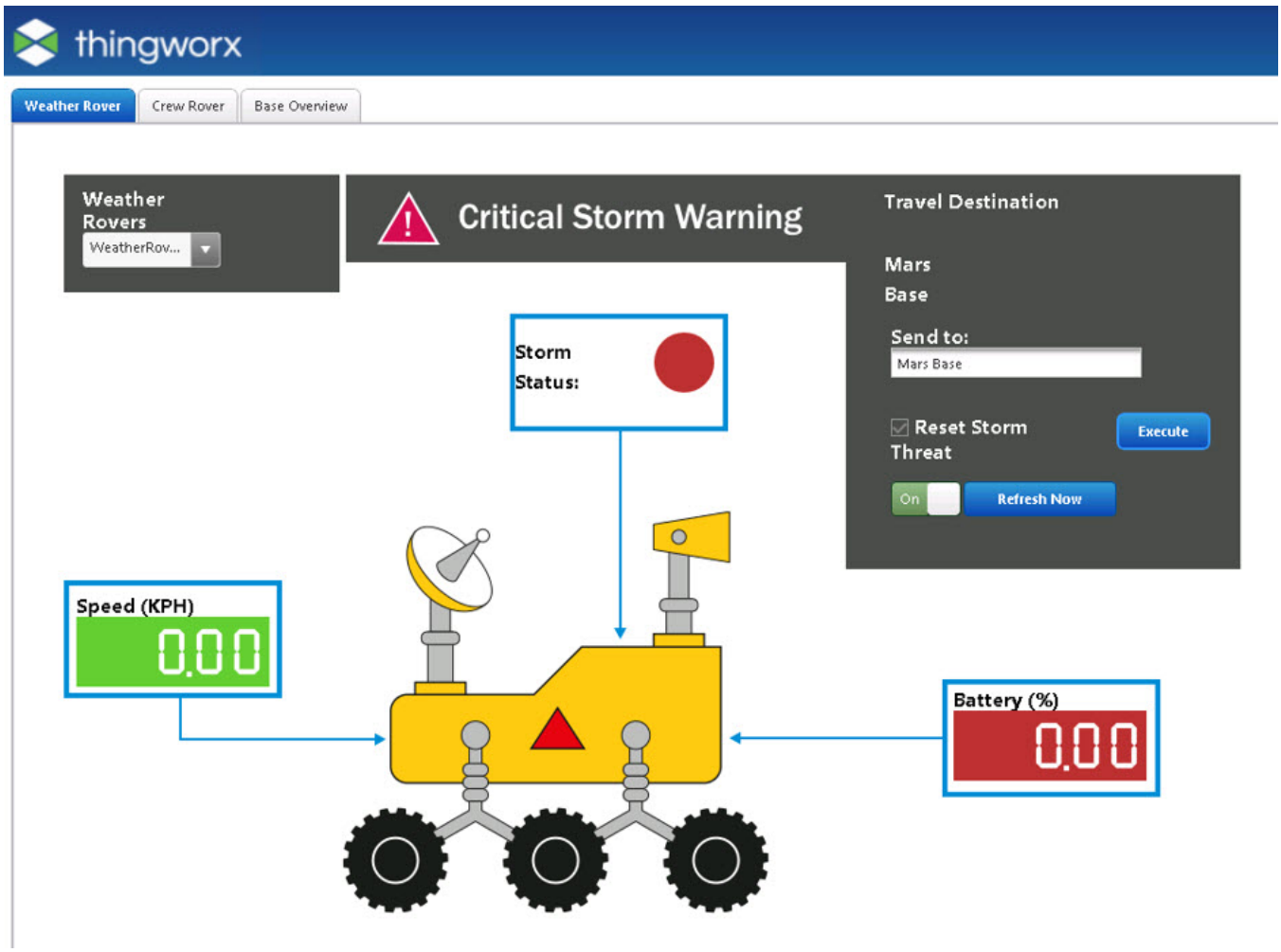
    ***Note**: The SetProperties service allows us to set the properties of our Thing dynamically from the Thing that is selected in the list widget. To set a property based on the selection from the list widget, we need to pass the name to the SetProperties service. To do this, we will use the Selected Row(s) property "name" of the GetImplementingThingsWithData service.*
29. From the ThingTemplates_WeatherRoverTemplate > GetmplementingThingsWithData > Returned Data > Selected Row(s) service, drag the **name** property to the EntityName property of the DynamicThingTemplates_WeatherRoverTemplate service.
30. Expand the DynamicThingTemplates_WeatherRoverTemplate > SetProperties service.

    ***Note:** Notice the list of all the properties that data can be bound to.*
31. Select the **Send To:** Textbox widget. Cursor over the arrow on the top left corner and drag **Text** to the SetProperties > Destination property.
32. Select the **Reset Storm Threat** checkbox widget (checkbox widget in the larger grey box area). Cursor over the arrow on the top left corner and drag **State** to the SetProperties > IsThreat property.
33. Select the **Execute** button widget. Cursor over the arrow on the top left corner and drag **Clicked** to the SetProperties service.

    ***Note**: The Clicked event triggers the SetProperties service.*
34. Click **Save**.
35. Click View Mashup.
36. In the Send to: textbox, type **Mars Base**.
37. Click the **Execute** button.

    ***Note**: The label widget below Travel Destination updates to Mars Base.*
38. If necessary, clear the Reset Storm Threat checkbox. Wait until the Storm Status shape turns green and the Critical Storm Warning image disappears.
39. Select the **Reset Storm Threat** checkbox.
40. Click the **Execute** button.

***Note**: The Storm Status (shape widget) changes to red and the Critical Storm Warning image appears.*



41. Close the **MarsMaster** tab to close the mashup view window.

M8:7 Crew Rover Widget Setup

1. Select the **Weather Rover** tab.
2. Press CTRL and select the following:
    - The Weather Rovers List widget.
    - The Critical Storm Warning Image.
    - The Speed LED.
    - The Battery LED.

- The Travel Destination and Destination Labels.
- The Send to: Textbox.
- The Execute button.
- The Auto Refresh widget.

3. With all the widgets selected, click the **Copy** icon located in the toolbar above the workspace.
4. On the left, select the **Workspace** tab.
5. Under Tab #2, select the **panel** where we are going to place these widgets. Make sure that the Crew Rover Tab is visible.

   ***Note**: You must not select the image, but the panel behind the image.*
6. Click the **Paste** icon located in the toolbar above the workspace.
7. Move the widgets around to align as necessary.
   - The Speed LED goes in the leftmost rectangle.
   - The Battery LED goes in the rightmost rectangle.
8. Select the **Weather Rovers** list widget.
9. In the Widget properties pane for the selected List widget, update the following property:

Property	Value
Label	Crew Rovers

10. In the left pane, click the Widgets tab.
11. Drag-and-drop an **LED Display** widget to the specified Current Passengers area in the top left rectangle.
12. In the Widget properties pane of the selected LED widget, update the following properties:

Property	Value
LEDLabelStyle	MarsMashupLabelStyle
Label	Current Passengers

    ***Note**: As there is an image on the mashup, you might not be able to place another widget on top of it directly. You will have to enhance the Preview of the Mashup to 1366x768, place the widget on the panel, and then cut and paste it onto the area specified on the mashup.*
13. Drag-and-drop an **LED Display** widget to the specified Max Passengers area in the topmost rectangle.
14. In the Widget properties pane of the selected LED widget, update the following properties:

Property	Value
LEDLabelStyle	MarsMashupLabelStyle
Label	Max Passengers

15. Drag-and-drop two **Button** widgets into the workspace.

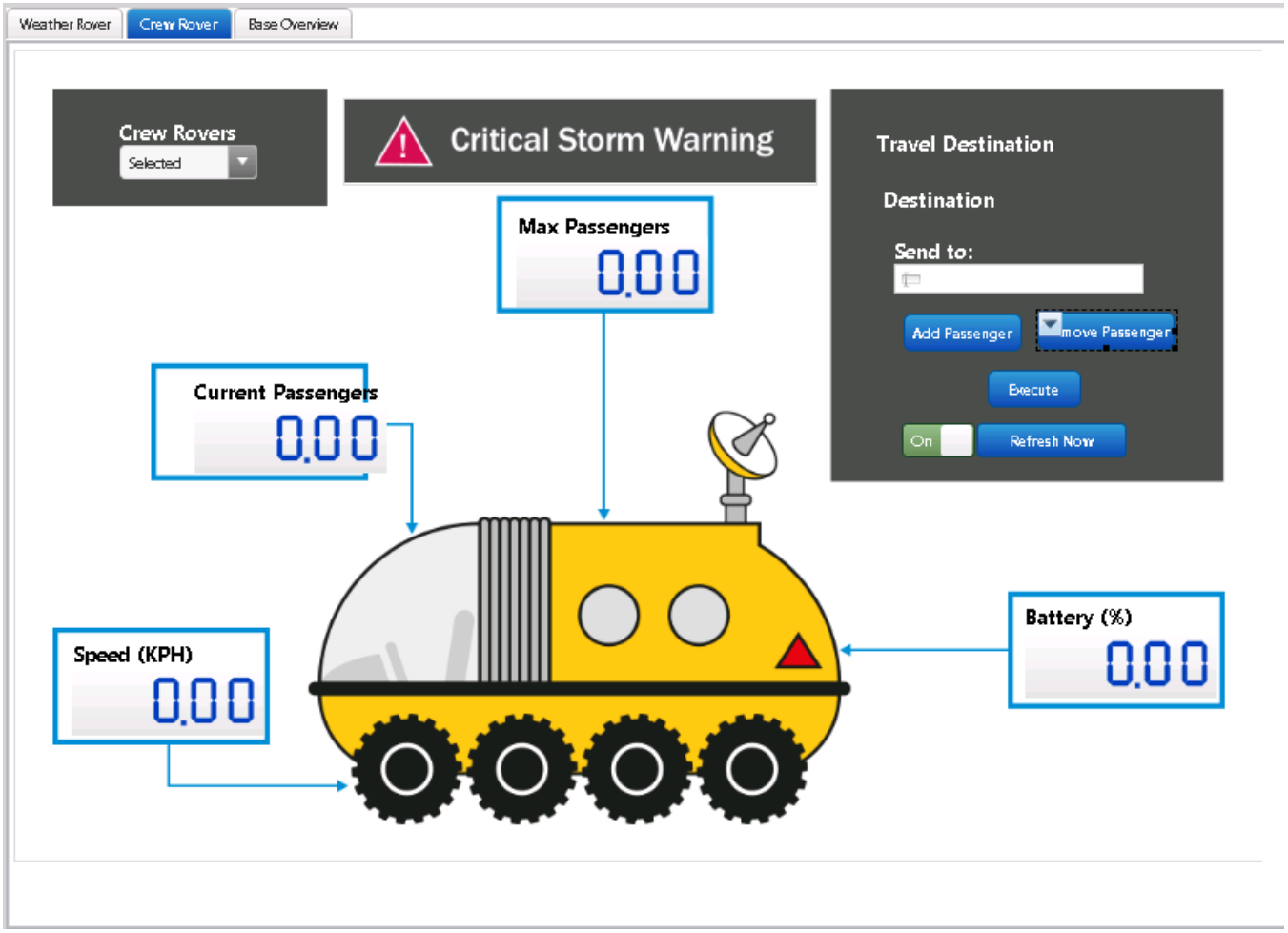
16. Position the Button widgets in the larger grey box side by side under the Send to: Checkbox. Move the Execute Button under them. Refer to the image below to see the button locations.
17. Select the Button widget on the left. In the Widget properties pane, update the following properties:

Property	Value
Label	Add Passenger
Top	170
Width	120

18. Select the Button widget on the right. In the Widget properties pane, update the following properties:

Property	Value
Label	Remove Passenger
Top	170
Width	120

19. Resize and arrange the widgets as necessary.



20. Click **Save** to save the mashup.

M8:8. Crew Rover Widget Binding

1. In the right pane, verify that the Data tab is selected.
2. Click the **green plus sign** icon on the top right of the Entities pane. An Add Data window appears.
3. In the Select Entity field, type/select **CrewRoverTemplate**.
4. In Select Services filter field, begin typing **GetImplementingThingsWithData** and click the **blue arrow** next to it.
5. Select the **Mashup Loaded?** check box.
6. Click **Done**.
7. In the ThingTemplates_CrewRoverTemplate > GetImplementingThingsWithData > Returned Data section, drag the **All Data** row to the Crew Rovers List widget. Select **Data** as the Binding Target.
8. In the Widget properties pane for the selected Crew Rovers List widget, update the following properties:

Property	Value
DisplayField	name
ValueField	name

9. Expand the ThingTemplates_CrewRoverTemplate > GetImplementingThingsWithData > Returned Data > Selected Row(s) section and drag the property to the widget using the binding target listed below:

Property	Widget	Binding Target
Speed	Speed LED Display	Data
Current Passengers	Current Passengers LED Display	Data
Max Passengers	Max Passengers LED Display	Data
Battery	Battery LED Display	Data
Destination	Destination Label	Text

10. Select the **Current Passengers** LED Display widget.
11. In the Widget properties pane of the selected LED Display widget, complete the following:
    - For the LEDFormatter property, click **State Formatting**.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **CurrentPassengers**.
    - In the State Definitions Search field, type/select **RedHighState**.
    - In the Value column, update the following values:

State	Value
Acceptable	< 2
Unfavorable	< 3

- Click **Done**.

12. Select the **Max Passengers** LED Display widget.
13. In the Widget properties of the selected widget, complete the following:
    - For the LEDFormatter property, click **State Formatting**.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **MaxPassengers**.
    - In the State Definitions Search field, type/select **GreenHighState**.
    - In the Value column, update the following values:

State	Value
Dangerous	< 1
Unfavorable	< 3

    - Click **Done**.
14. Expand the ThingTemplates_WeatherRoverTemplate > GetImplementingThingsWithData > Returned Data > Selected Row(s) section. Drag the **IsThreat** property to the Critical Storm Warning image widget and select **Visible** as the Binding Target.
15. Cursor over the arrow on the top-left corner of the Auto Refresh widget and drag the **Refresh** event to the **ThingTemplates_CrewRoverTemplate > GetImplementingThingsWithData** service.
16. Click the green **plus sign** icon on the top right of the Data tab. An Add Data window appears.
17. Select the **Dynamic** check box.
18. In the Select Entity field, type/select **CrewRoverTemplate**.
19. In Select Services filter field,
    - Begin typing **SetProperties** and click the **blue arrow** next to it.
    - Begin typing **AddPassenger** and click the **blue arrow** next to it.
    - Begin typing **RemovePassenger** and click the **blue arrow** next to it.
20. Click **Done**.
21. From the ThingTemplates_CrewRoverTemplate > GetmplementingThingsWithData > Returned Data > Selected Row(s), drag the **name** property to the EntityName property of the DynamicThingTemplates_CrewRoverTemplate service.
22. Expand the DynamicThingTemplates_CrewRoverTemplate > SetProperties service.

23. Select the **Execute** button widget. Cursor over the arrow on the top-left corner and drag **Clicked** to bind it to the SetProperties service.
24. Select the **Send To:** Textbox widget. Cursor over the arrow on the top left corner and drag **Text** to the SetProperties > Destination property.
25. Select the **Add Passenger** Button widget. Cursor over the arrow on the top left corner and drag **Clicked** to the SetProperties > AddPassenger service.
26. Select the **Remove Passenger** Button widget. Cursor over the arrow on the top left corner and drag **Clicked** to the SetProperties > RemovePassenger service.
27. Click **Save**.

M8:9. Base Overview Tab

1. Select the **Crew Rover** tab in the mashup's Design tab.
2. Press CTRL and select the following:
    - The Critical Storm Warning Image.
    - The Current Passengers LED.
    - The Auto Refresh widget.
3. With all the widgets selected, click the **Copy** icon located in the toolbar above the workspace.
4. On the left, select the **Workspace** tab.
5. Under Tab #3, select the **panel** where we are going to place these widgets. Make sure that the Base Overview Tab is visible.

    ***Note**: You must not select the image, but the panel behind the image.*
6. Click the **Paste** icon located in the toolbar above the workspace.
7. Move the widgets around to align as necessary.
    - Drag the **Critical Storm Warning** Image widget below the Main Base.
    - Drag the **LED Display** widget to the specified North Station area as on the image.
    - Drag the **Auto Refresh** widget to the gray box in the top right.

    ***Note**: The image is at the end of this exercise.*
8. Verify that the ThingTemplates_WeatherRoverTemplate > GetImplementingThingsWithData > Returned Data > Selected Row(s) section is expanded, if not, then expand it.
9. Drag the **IsThreat** property to Critical Storm Warning image widget and select **Visible** as the Binding Target.

10. Click the green **plus sign** icon on the top right of the Data tab. An Add Data window appears.
11. In the Select Entity field, type/select **NorthWeatherStation**.
12. In the Select Services filter field,
    - Begin typing **GetProperties** and click the **blue arrow** next to it.
    - Select the **Mashup Loaded?** check box.
13. Reset the Select Entity field by clicking the **x** icon next to NorthWeatherStation.
14. In the Select Entity field, type/select **EastWeatherStation**.
15. In the Select Services filter field,
    - Begin typing **GetProperties** and click the **blue arrow** next to it.
    - Select the **Mashup Loaded?** check box.
16. Repeat these steps for the following Entities:
    - SouthWeatherStation
    - WestWeatherStation
17. Click **Done**.
18. Expand the **Things_NorthWeatherStation** > **GetProperties** service. Drag the **WindSpeed** property to the North Weather Station LED Display widget and select **Data** as the Binding Target.
19. Select the **LED Display** widget at North Station.
20. In the Widget properties pane for the selected LED Display widget, update the following property:

Property	Value
Label	North Station

21. In the Widget properties pane of the selected LED Display widget, complete the following:
    - For the LEDFormatter property, hover over the right field, then click **RESET**.
    - Click the **State Formatting** button.
    - Select the **State-based Formatting** radio button.
    - From the Dependent Field drop-down list, select **WindSpeed**.
    - In the State Definitions Search field, type/select **RedHighState**.
    - In the Value column, update the following values:

State	Value
Acceptable	< 65
Unfavorable	< 75

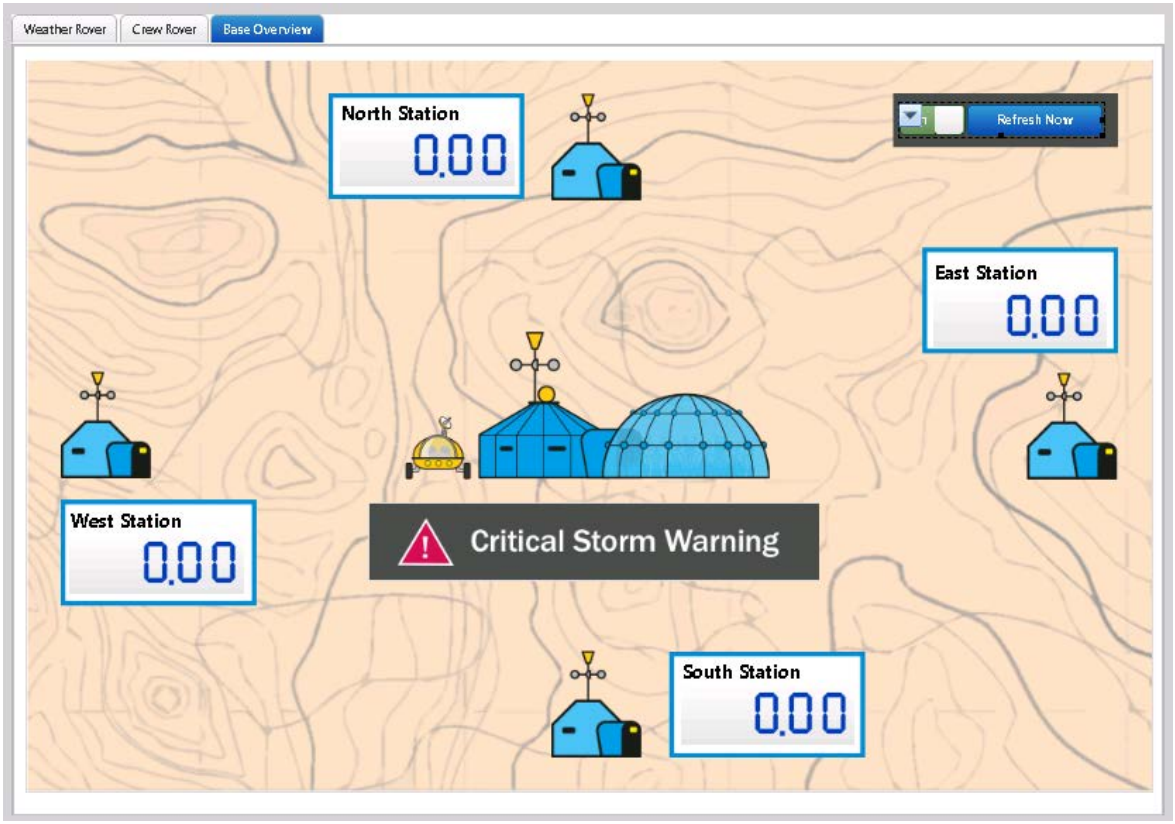
- Click **Done**.

22. Select the **LED Display** widget at North Station.
    - Click the **Copy** icon located in the toolbar above the workspace.
    - In the Workspace tab, under Tab #3, select the **panel**.
    - Click the **Paste** icon located in the toolbar above the workspace.
    - Drag the **LED Display** widget to the specified East Station area as on the image.
23. Expand the **Things_EastWeatherStation** > **GetProperties** service. Drag the **WindSpeed** property to the East Weather Station LED Display widget and select **Data** as the Binding Target.
24. Select the **LED Display** widget at East Station.
25. In the Widget properties pane for the selected LED Display widget, update the following property:

Property	Value
Label	East Station

26. Paste the LED widget two more times, position them, bind the appropriate WindSpeed property, and and update the labels for the following stations:
    - South Station
    - West Station
27. Hover over the **Auto Refresh** widget near the upper left corner.
28. Select **Refresh** and drag it to the data pane on the right under **Things_NorthWeatherStation > GetProperties**.
29. Hover over the Auto Refresh widget, drag Refresh to Things_EastWeatherStation > GetProperties.
30. Repeat binding the **Auto Refresh** widget to **GetProperties** under the following things:
    - Things_SouthWeatherStation
    - Things_WestWeatherStation

31. Click **Save**.



M8:10. Testing the Application

1. Click the **View Mashup** button at the top.
2. Set the Initial conditions for the Weather Rovers. In the Weather Rover tab, do the following:
    - Make sure the Dust Storm is not enabled already, if it is, clear the **Reset Storm Threat** check box and click **Execute**.
    - In the Send to: field type **Mars Base** and then click **Execute** to send your Rover to Mars Base.
2. Set the Initial conditions for the Crew Rovers. Select the **Crew Rover** tab.
    - Select **CrewRover1** from the CrewRover drop-down list. In the Send to: field, type **Mars Base** and then click **Execute** to send CrewRover1 to Mars Base.
    - Select **CrewRover2** from the CrewRover drop-down list. In the Send to: field, type **Mars Base** and then click **Execute** to send CrewRover2 to Mars Base.
    - Select **CrewRover3** from the CrewRover drop-down list. In the Send to: field, type **Mars Base** and then click **Execute** to send CrewRover3 to Mars Base.

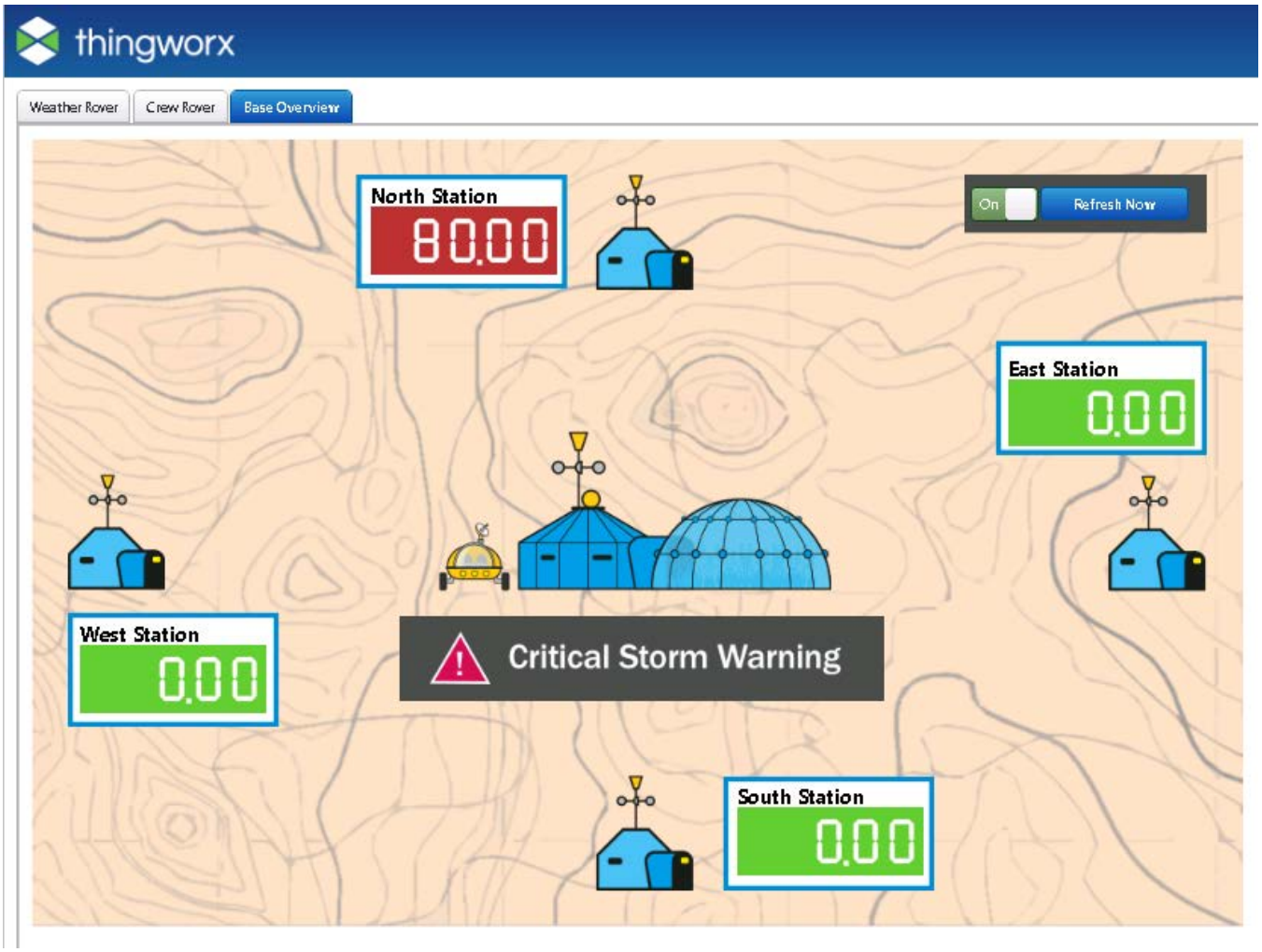
3. Set the Initial conditions for the Base Overview. Select the **Base Overview** tab.
    - Ensure that none of the Weather Stations are in their "Dangerous condition" which means that their WindSpeed is above 75.
    - If the WindSpeed of any of the Weather Stations is above 75 then navigate back to the ThingWorx Composer. Open that specific Weather Station thing in the Composer and change the WindSpeed property of that weather station thing to **30**.
4. Return to the Composer tab in Chrome.
5. In Chrome, copy the ThingWorx URL up to and including **/Thingworx**.
6. Paste this URL into a document in a text editor, such as Notepad.
7. Open the **ColonyKey** Application Key entity.
8. If the selected date in the Expiration Date field is already expired, then select a **date one week from now**.
9. Copy and paste the **Key ID** of the ColonyKey into the text editor document.
10. Open Postman.
11. In Postman, ensure that the http verb to the left of the Enter request URL field is set to **PUT** from the drop-down list.
12. In the Enter request URL field, type the URL that follows. Replace the bracketed text with the URL you copied to your text editor earlier. Replace *username* with your user name. Replace NorthWeatherStation and/or Windspeed if your spelling is different.

    [copied URL]/Things/NorthWeatherStation_*username*/Properties/WindSpeed
13. Select the **Headers** tab.
14. In the New key field, type **appKey**.
15. In the Value field, paste your Application Key (**Key ID**) from the ColonyKey.
16. On the next line, in the New key field, type/select **Content-Type**.
17. In the Value field, type/select **application/json**.
18. Select the **Body** tab to the right of the Headers tab.
19. Select the **raw** radio button.
20. Ensure that your formatting is JSON using the drop-down list on the right.
21. In the text editing area, type:

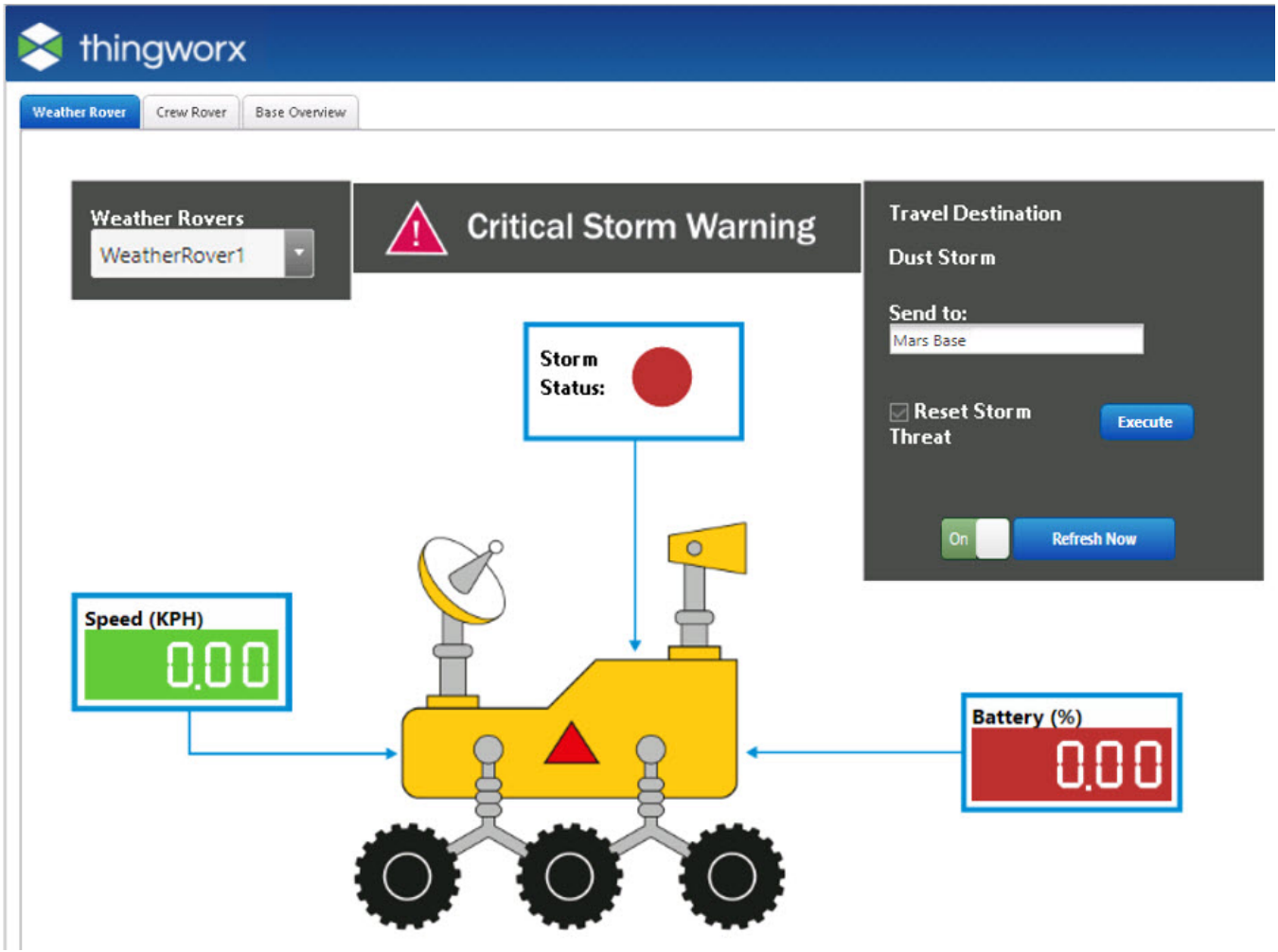
```
{"WindSpeed":"80" }
```

22. Click **Send**.
23. Select the **MarsMaster** tab in Chrome.

24. Select the **Base Overview** tab. Notice you will see the North Station display 80 and your state-based formatting changed the style to Red.

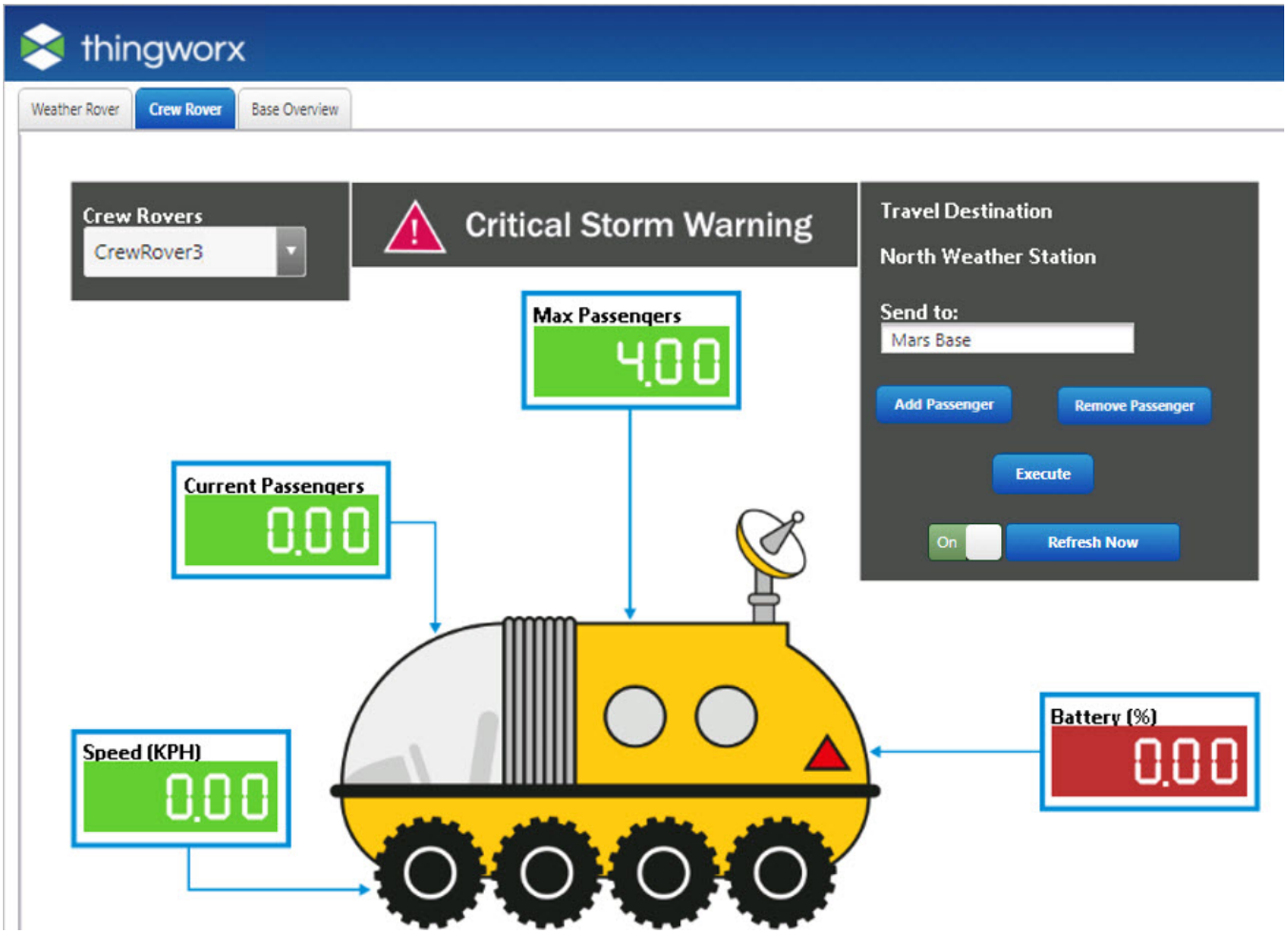


25. Select the **Weather Rover** tab.
    - Notice that the Travel Destination is now set to “Dust Storm”.
    - Notice the warnings to indicate that a dangerous storm has occurred.



26. Select the **Crew Rover** tab.

- Notice that the destination has been set to North Weather Station.



27. As an optional task, use Postman, or the Academic Edge Connector, to set the Speed and Battery properties to various values and verify they update accordingly in the tabs.
28. As an optional task, use Postman, or the Academic Edge Connector, to add and remove passengers to the Crew Rovers and verify they update. Also verify they do not exceed 4 passengers and the LEDs change colors.
29. As an optional task, update the LED widget properties to only display 2 digits and no decimals.