



thingworx

Interstacks
Edge Device Starter Kit

Interstacks Edge Device Starter Kit

Copyright © 2017 PTC Inc. and/or Its Subsidiary Companies. All Rights Reserved.

User and training guides and related documentation from PTC Inc. and its subsidiary companies (collectively "PTC") are subject to the copyright laws of the United States and other countries and are provided under a license agreement that restricts copying, disclosure, and use of such documentation. PTC hereby grants to the licensed software user the right to make copies in printed form of this documentation if provided on software media, but only for internal/personal use and in accordance with the license agreement under which the applicable software is licensed. Any copy made shall include the PTC copyright notice and any other proprietary notice provided by PTC. Training materials may not be copied without the express written consent of PTC. This documentation may not be disclosed, transferred, modified, or reduced to any form, including electronic media, or transmitted or made publicly available by any means without the prior written consent of PTC and no authorization is granted to make copies for such purposes.

Information described herein is furnished for general information only, is subject to change without notice, and should not be construed as a warranty or commitment by PTC. PTC assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

The software described in this document is provided under written license agreement, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. It may not be copied or distributed in any form or medium, disclosed to third parties, or used in any manner not provided for in the software licenses agreement except with written prior approval from PTC.

UNAUTHORIZED USE OF SOFTWARE OR ITS DOCUMENTATION CAN RESULT IN CIVIL DAMAGES AND CRIMINAL PROSECUTION. PTC regards software piracy as the crime it is, and we view offenders accordingly. We do not tolerate the piracy of PTC software products, and we pursue (both civilly and criminally) those who do so using all legal means available, including public and private surveillance resources. As part of these efforts, PTC uses data monitoring and scouring technologies to obtain and transmit data on users of illegal copies of our software. This data collection is not performed on users of legally licensed software from PTC and its authorized distributors. If you are using an illegal copy of our software and do not consent to the collection and transmission of such data (including to the United States), cease using the illegal version, and contact PTC to obtain a legally licensed copy.

Important Copyright, Trademark, Patent, and Licensing Information: See the About Box, or copyright notice, of your PTC software.

UNITED STATES GOVERNMENT RESTRICTED RIGHTS LEGEND

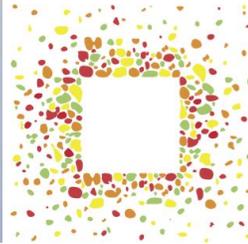
This document and the software described herein are Commercial Computer Documentation and Software, pursuant to FAR 12.212(a)-(b) (OCT'95) or DFARS 227.7202-1(a) and 227.7202-3(a) (JUN'95), and are provided to the US Government under a limited commercial license only. For procurements predating the above clauses, use, duplication, or disclosure by the Government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 (OCT'88) or Commercial Computer Software-Restricted Rights at FAR 52.227-19(c)(1)-(2) (JUN'87), as applicable. 01012015

PTC Inc., 140 Kendrick Street, Needham, MA 02494 USA



Introduction - About Interstacks	3
Installation - Download Stackbuilder	4
Configuration and Setup	5
Make a new project blueprint	5
Configure and Run with Thingworx	8
Technical Notes for more advanced users	11
Troubleshooting	12
Compatibility	12
ThingWorx Platform Version	12
Document Revision History	13

Introduction - About Interstacks



INTERSTACKS

Interstacks is a system of snap-together, modular electronic blocks and visual authoring tool (Stackbuilder), that enables anyone to rapidly build and deploy new IoT edge hardware.

- First choose the electronic hardware blocks you want to use for a project.
- Snap them together into a stack.
- Download and install the Stackbuilder authoring tool.
- Connect the included USB cable between your stack and computer.
- Connect power to your stack using the included wall power supply.
- Use the Stackbuilder software to create a stack blueprint that tells your stack what you want it to do.
- Press the "Send to Stack" button in Stackbuilder to send your blueprint to your connected stack and save it in the stack.

After your blueprint is working correctly, you can disconnect the stack from your computer. Every time your stack is connected to power, it will load the blueprint you sent to it.

The Stackbuilder authoring tool allows you to create a stack blueprint ("program a stack") by simply dragging and dropping hardware and software blocks into the blueprint area, and then drawing lines (channels) between the input and output terminals of those blocks.

You can mix hardware and software blocks in your blueprint. The hardware blocks correspond to the Interstacks hardware you purchased. Software blocks can be found in the built-in "Software" tab in Stackbuilder or you can create your own, in the "My Blocks" tab of Stackbuilder, by typing text using the **Python programming** language.

Please refer to documentation on many topics at <http://interstacks.com/support>.

Please email info@interstacks.com with any questions.

Installation - Download Stackbuilder

Stackbuilder is the visual authoring tool that allows you to tell your stack what to do by making stack "blueprints" and sending them to your stack via a USB cable connection. You need to download Stackbuilder for your Mac or PC by going to interstacks.com/stackbuilder and clicking on the appropriate download button for your computer.

Please follow the standard installation instructions for your Mac or PC. Once installed on a Mac, you may have to hold down the "control" key while clicking on the Stackbuilder icon, then select "Open" from the pop up menu.

When you launch Stackbuilder for the first time, you will need to create an Interstacks account by entering your email address and a password. Your work can then be optionally stored in the cloud for access from any computer using your login. You will be sent a confirmation email from interstacks.com with the subject "Verify your Interstacks Account". You will need to click on the "CLICK HERE" link to verify your email address before your Interstacks account becomes valid.

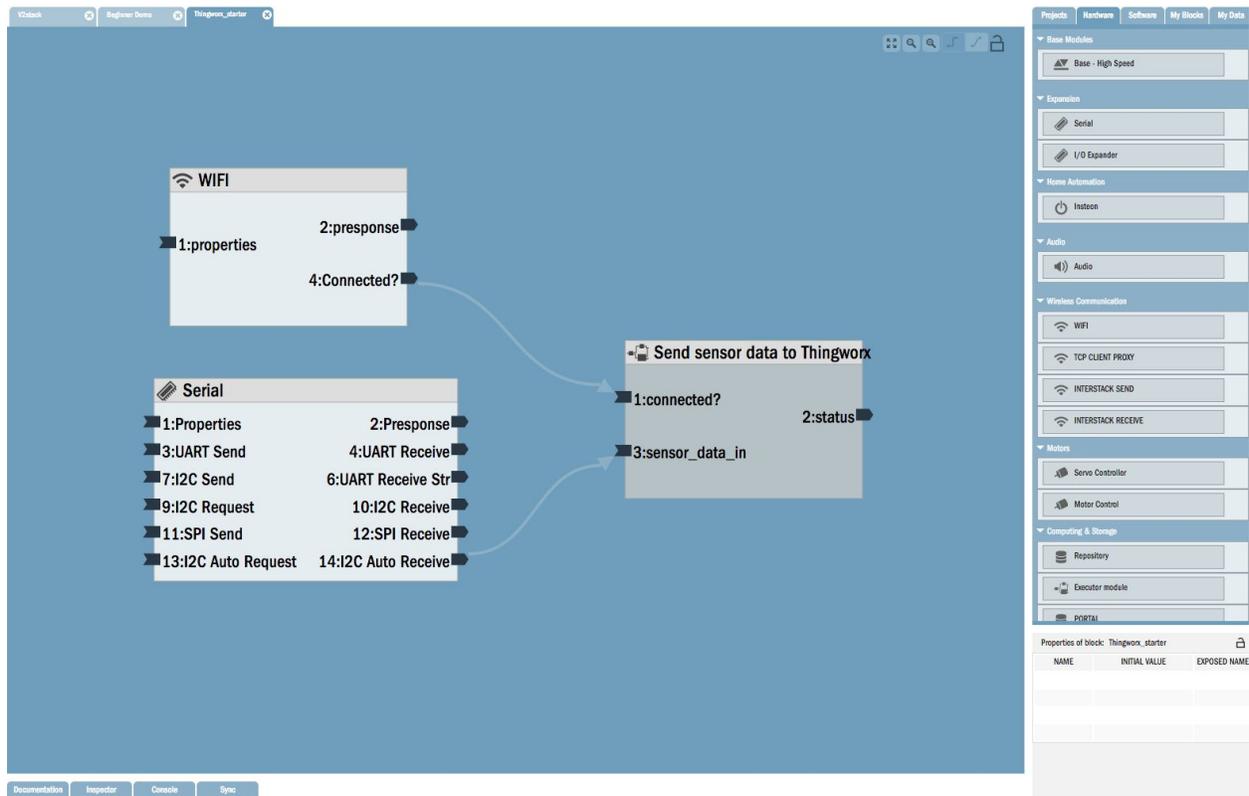
After you launch Stackbuilder and login to your account, your projects are typically synced to your cloud account so that you have the option of developing on more than one computer and have your work backed up. If this is not desired or you don't have an active network connection, simply click on the "Skip" button. After syncing your projects, the Stackbuilder interface will be displayed.

Windows USB Driver: The first time a stack is connected via USB cable to a Windows computer, it is sometimes necessary to manually assign a USB driver. After this is done the first time, it will not be necessary again. You will know you need to do this if the "Send to Stack" button does not appear in the Stackbuilder interface after you connect your stack to your computer.

Upon connecting a stack via USB, the System Tray will indicate searching for the USB driver. If it indicates it couldn't find the driver, click on the selection it presents. The driver is assigned via the Control Panel -> Device Manager -> USB Devices. In the tree list, find isxusb, right click and select "update driver". Then select "Auto Update". If that doesn't assign a driver, select your own via "Universal Serial Bus" and select "Win USB".

There is detailed documentation on installing Windows USB drivers for Interstacks online at <http://interstacks.com/support/windows-usb>.

Configuration and Setup



Make a new project blueprint

Please refer to documentation on many topics at <http://interstacks.com/support>. Please email info@interstacks.com with any questions.

The Stackbuilder authoring tool allows you to create a stack blueprint (“program a stack”) by simply dragging and dropping hardware and software blocks into the blueprint area, and then drawing lines (channels) between the input and output terminals of those blocks. Each block has input and output terminals that send or receive messages. Messages can be numbers, characters, strings, binary data, or other data structures like lists (e.g. [2, 4, 6, 8]) and Python dictionaries. The channels you draw define where messages sent from output terminals are delivered. You can draw as many channels in to, or out of, a terminal as you want. In this way, you are defining the asynchronous message flow that will run inside your stack. By allowing multiple stacks to communicate with each other and/or the internet, you can build large scale distributed systems.

You can mix hardware and software blocks in your blueprint. The hardware blocks correspond to the Interstacks hardware modules you purchased. Software blocks can be found in the built-in “Software” tab in Stackbuilder or you can create your own, in the “My Blocks” tab of Stackbuilder, by typing text using the Python programming language. The

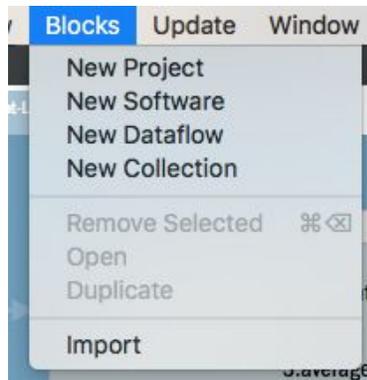
Interstacks Edge Device Starter Kit

Stackbuilder interface has four primary areas:

- **Blueprint:** Main authoring area that displays message flows or the software block editor (Python programming).
- **Right rail:** Blocks to drag and drop into the blueprint.
- **Property Editor:** To edit properties for a selected component.
- **Bottom rail:** For context sensitive documentation and the Console trace for message flow debugging.

All of these areas have tabs and can be expanded or closed to manage your workspace. All project edits are immediately saved to your local storage and to your Interstacks cloud account if a network connection is available.

To start a new project with a blank blueprint, **select “New Project”** from the “Blocks” pull down menu in the Application’s menu bar.



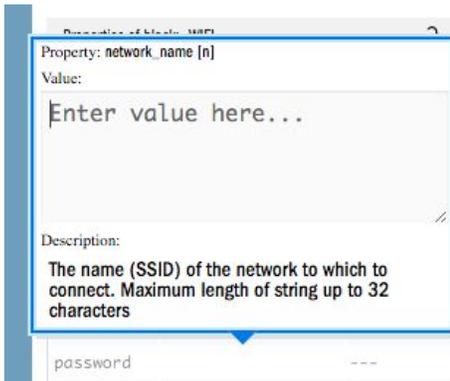
You will see a new entry in the “Projects” palette called “Unnamed Project”. To rename it, click in the text area and edit the name. To open the new project, **double click in the rectangular box** containing its name.

This starter project assumes you have the Interstacks IoT starter stack, or have a Serial hardware block connected to some sensor and a WiFi hardware block for internet communication. We will stream the sensor data to the Thingworx platform using HTTP REST communication. This project uses a luminosity light sensor connected via the I2C serial communications protocol to an Interstacks Serial hardware block.

After you have created and opened your new blank blueprint, drag and drop the Wifi module and Serial module from Stackbuilder’s right rail, “Hardware” tab, onto the blank blueprint area of your new project.



Interstacks Edge Device Starter Kit



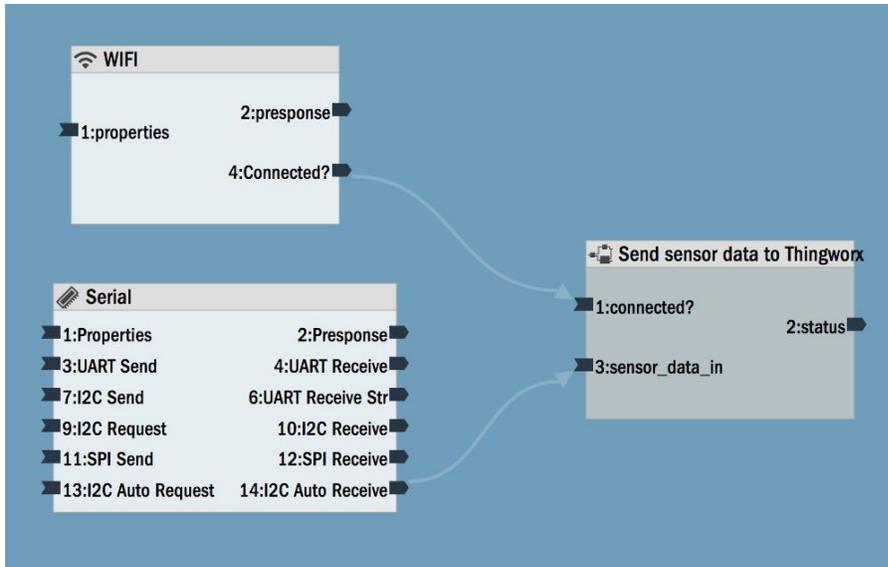
Select the Wifi module in your blueprint by clicking on it once. Use the property editor in the bottom right to set the LAN network name the Wifi module will connect to. Click in the "Initial Value" column for the network_name property. You will see a popup that lets you type the name e.g. "mynetwork". You need to type the quotes at the beginning and end of the name. Click in the "Initial Value" column for the password property. Enter the LAN password, again typing the quotes at the beginning and end e.g. "mypassword".

Now select the Serial module by clicking on it once. Set the i2c_rate property in the property editor to 1 by clicking in the Initial Value column. This will cause the light sensor value to be sent every second out of the "I2C Auto Receive" terminal of the Serial block if the value has changed.

Next, drag and drop the **"Send sensor data to Thingworx"** block into your blueprint from the "Software" tab in the right rail "Network & Communication" hierarchy. There are five potential properties that can be set for this block: **appkey, port, server, thing name, and thing property name**. Use the property editor to set the server name URL (or IP address) for your Thingworx server e.g. "<http://thingworx.myco.com>". Set the thing name string e.g. "Demo_Light_Sensor". Set the thing property name e.g. "lightsensor". Lastly, you must set the App Key property to the authentication key you generated from your Thingworx server. Again, click on the "Initial Value" column for the App Key property and enter your key e.g. "fee9e81f-c733-4e5f-9f12-767c829b7777". Make sure you type the quotes for all string values.

Now **draw two channels**: one from the "Connected?" output terminal of the Wifi block to the "Connected?" input terminal of the Thingworx block, and one from the "I2C Auto Receive" output terminal of the Serial block to the "sensor_data_in" input terminal of the Thingworx block. Your blueprint should now look like this:

Interstacks Edge Device Starter Kit



Configure and Run with Thingworx

While the purpose of this document is not to be a Thingworx Composer user manual, we will proceed in giving some guidelines and relevant details for this project.

To view the sensor data from your Interstacks stack, you need to set up your Thingworx server mashup. Log in to your Thingworx server instance in your normal way. Probably via a URL of the general form:: <http://myserver.myco.com/Thingworx/Composer/> . Create a new thing in your composer, name it e.g. "Demo_Light_Sensor". Use the "GenericThing" thing template (since we are using the HTTP REST protocol, not the AlwaysOn protocol). Create a stream using the "Valuestream" template. Go back to the "Demo_LLght_Sensor" thing and select that stream for the Valuestream field. Now add a property to your thing. Give it a name, check the "Logged" check box, and set the "Base Type" to "#Number". This is so that the sensor data values will be recorded. Following are some examples from Thingworx Composer "General Information" and "Properties" forms:

Interstacks Edge Device Starter Kit

The screenshot shows the ThingWorx interface for editing the 'Demo_Light_Sensor' entity. The 'General Information' tab is active, displaying various fields for configuration. The left sidebar contains navigation options like 'General Information', 'Properties', 'Services', 'Events', 'Subscriptions', 'Home Mashup', 'PERMISSIONS', 'CHANGE HISTORY', and 'DEPENDENCIES'. The main content area includes fields for Name, Description, Project, Tags, Thing Template, and Implemented Shapes. On the right, there are checkboxes for 'Active' and 'Published', a 'Home Mashup' field, an 'Avatar' field, an 'Identifier' field, a 'Last Modified Date' field, and a 'Value Stream' field.

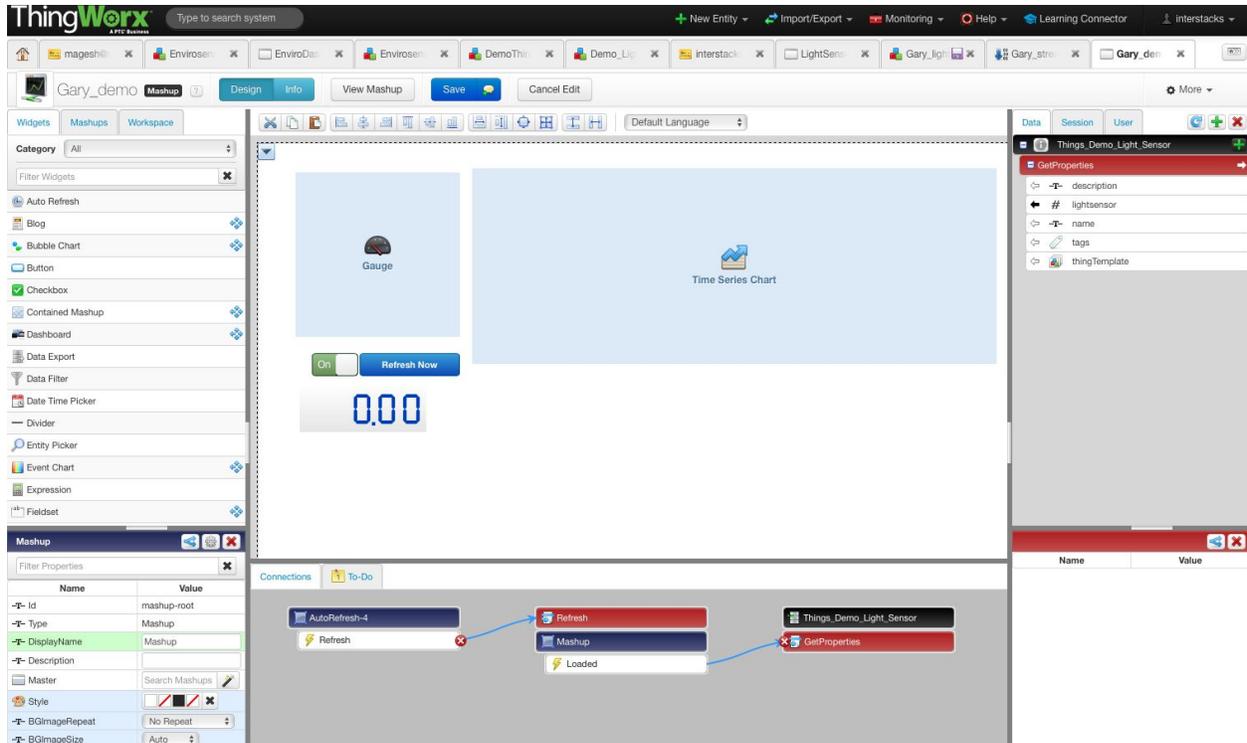
The screenshot shows the ThingWorx interface for editing the 'Gary_light_sensor' entity. The 'Properties' tab is active, displaying a table of 'My Properties' and a 'New Property' form. The table has columns for Edit, Name, Type, Alerts, Additional Info, Default Value, Value, and DataChange. The 'New Property' form includes sections for 'General Property Info', 'BaseType Info', 'Alerts', 'Aspects', and 'Data Change Info'. The 'General Property Info' section has fields for Name, Description, and Category. The 'BaseType Info' section has a 'Base Type' dropdown set to '# NUMBER', and fields for Units, Min Value, Max Value, and Has Default Value. The 'Aspects' section has checkboxes for Persistent, Read-only, and Logged. The 'Data Change Info' section has a 'Data Change Type' dropdown set to 'Value' and a 'Change Threshold' field.

Get your application key from the Composer Home - View All. Select the entry with the type "Application Key" from the list. That key goes into your Interstacks blueprint as described previously (property for the "Send sensor data to Thingworx" block). If you have not created a key before, look at the New Entity pull down menu to create a key.

Now use Thingworx Composer to create a mashup. Go to Home. Select the "+" next to Mashup. Edit the Mashup and add the Widgets you want. For example the Gauge widget. Add the data stream to the "Data" tab on the far right. Hit the + next to that data stream, drag and drop the #lightstream entry on top of the Gauge widget (bind it to the "Data" when prompted). Our demonstrations also use the LED and Timeseries widgets but you should use what you

Interstacks Edge Device Starter Kit

prefer. Drag the Auto Refresh widget into your Mashup, and make sure the refresh rate is set to 1 (second) and the appropriate Refresh checkboxes are selected. Also make sure the “Loaded” event for the Mashup is added. Click the “View Mashup” button to launch your Mashup. See our example Mashup below:

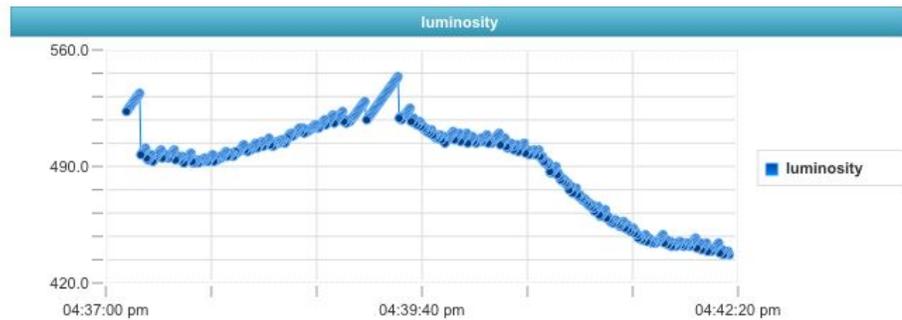


Connect power to your stack. Connect the USB cable from your stack to your computer. Send the blueprint you built previously to your stack by clicking the **“Send to Stack”** button in the top middle of your blueprint. (If that button does not appear, your stack is not communicating with your computer. If on Windows, see the Windows USB drivers documentation mentioned in the Install instructions above).

After your blueprint is loaded in your stack, the bottom rail will display a console trace of the messages flowing inside your stack. (You can close this by clicking on the white space between the bottom rail and the blueprint). Once your blueprint is fully functional, you can disconnect it from your computer. Every time the stack is powered up, it will load your blueprint. The blueprint will run somewhat faster when not connected to your computer and Stackbuilder.

Your light sensor data should now be viewable on your Thingworx Mashup. Congratulations! Below is an example Mashup using three Widgets bound to the stack light sensor data stream.

449.00



Technical Notes for more advanced users

Now that your first project is running with Interstacks and Thingworx, you should browse the online documentation at <http://interstacks.com/support>. This project uses Thingworx REST API calls to communicate with a Thingworx server. A quick overview of the Thingworx REST API is here: <http://community.thingworx.com/docs/DOC-3315>. The Interstacks Stackbuilder authoring tool has a HTTP API interface block in the “Software” tab of the right rail in the “Network & Communication” hierarchy (see online documentation <http://interstacks.com/support/internet-http>). The above project uses the “Set Property Values” Thingworx API. It uses PUT with URL:

/Things/{ThingName}/Properties/{propertyName} and Content-Type:"application/json" with the PUT Body: { {name of property}:{value of property} }. You can configure your Thingworx server to use HTTP or HTTPS. In the URL property of the Interstacks HTTP API block simply use the appropriate URL.

You can create your own software blocks that will show up in the “My Blocks” tab by using the “Blocks-> New MyBlocks Software” pull down menu entry. You can then write Python software in your own Software blocks (see <http://interstacks.com/support/make-software> and other online support documents).

For reference: a typical full HTTP PUT to send a data value in the above project is:

```
('PUT', 'http://54.173.93.7/Thingworx/Things/Demo_Light_Sensor/Properties/*', 'appKey: fee9e81f-c733-4e5f-9f12-767c829b7777\r\nAccept: */*\r\nUser-Agent: python-requests/2.8.0\r\nConnection: Close\r\nContent-Type: application/json;
```

Interstacks Edge Device Starter Kit

```
charset=utf-8\r\nContent-Length: 20\r\n\r\n{"lightsensor": 617})
```

This is a typical raw response from the Thingworx server. The status 200 indicates success.

```
('HTTP/1.1 200 OK', [['Server', 'Apache-Coyote/1.1'], ['X-FRAME-OPTIONS', 'SAMEORIGIN'], ['Expires', '0'], ['Cache-Control', 'no-store, no-cache'], ['Cache-Control', 'post-check=0, pre-check=0'], ['Pragma', 'no-cache'], ['Content-Type', 'text/html; charset=UTF-8'], ['Transfer-Encoding', 'chunked'], ['Date', 'Fri, 10 Mar 2017 20:00:28 GMT'], ['Connection', 'close']], bytearray(b'0\r\n\r\n'))
```

Troubleshooting

	Problem	Solution(s)
1.	"Send to Stack" button does not appear when stack is connected.	<ol style="list-style-type: none">1. Make sure USB cable is fully connected between stack and computer and that the stack is connected to power.2. If using a Windows computer, make sure USB driver is installed and selected correctly. see http://interstacks.com/support/windows-usb
2.	Data from sensor not showing up in Thingworx	<ol style="list-style-type: none">1. Verify the correct Appkey is being used in Thingworx and the Interstacks blueprint property AppKey for the "Send sensor data to Thingworx" block.2. Make sure Thingworx Composer Thing name and Thing property name are the same as the Interstacks blueprint properties for the "Send sensor data to Thingworx" block.3. Make sure the Server property in the Interstacks blueprint for the "Send sensor data to Thingworx" points to your Thingworx server instance.4. Verify your LAN network name and password are in the properties for you WiFi block in your Interstacks blueprint.

Compatibility

This guide has been tested for compatibility with Interstacks and the following ThingWorx platform and operating system:

ThingWorx Platform Version	ThingWorx 7.1.0
----------------------------	-----------------

Document Revision History

Revision Date	Version	Description of Change
March 8, 2017	1.0	First Release