

PI Connector for UFL REST endpoint samples

0.1

OSIsoft, LLC

777 Davis St., Suite 250
San Leandro, CA 94577 USA

Tel: (01) 510-297-5800

Fax: (01) 510-357-8136

Web: <http://www.osisoft.com>

Copyright: © 1992-2012 OSIsoft, LLC. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of OSIsoft, LLC.

OSIsoft, the OSIsoft logo and logotype, PI Analytics, PI ProcessBook, PI DataLink, ProcessPoint, PI Asset Framework (PI AF), IT Monitor, MCN Health Monitor, PI System, PI ActiveView, PI ACE, PI AlarmView, PI BatchView, PI Coresight, PI Data Services, PI Event Frames, PI Manual Logger, PI ProfileView, PI WebParts, ProTRAQ, RLINK, RtAnalytics, RtBaseline, RtPortal, RtPM, RtReports and RtWebParts are all trademarks of OSIsoft, LLC. All other trademarks or trade names used herein are the property of their respective owners.

U.S. GOVERNMENT RIGHTS

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the OSIsoft, LLC license agreement and as provided in DFARS 227.7202, DFARS 252.227-7013, FAR 12.212, FAR 52.227, as applicable. OSIsoft, LLC.

Contents

Sending data to the REST endpoint

- Setup
- Header and data format

Prerequisites to run samples

Value sample

- Python
 - Module(s) required
 - File(s) required
 - Procedure

Values sample

- Python
 - Module(s) required
 - File(s) required
 - Procedure

Sending data to the REST endpoint

This section provides information on sending data to the REST endpoint of the PI Connector for UFL. The following sections provide details on samples that include configuration files and scripts to send data files to the REST endpoint. The Connector can create Asset Framework (AF) objects and PI Data Archive tags as defined by the configuration file. The Connector also supports data input from files and serial ports which is not covered in this document.

Setup

1. PI Connector for UFL:
 - i. is installed and service started.
 - ii. is started (see admin GUI: <https://<server>:<port>/admin/ui>) where server is the host where the Connector is installed and port is the port selected on installation.
 - iii. is configured with a PI and AF data server.
2. Configure a PI Connector for UFL data source, specifying:
 - i. Data source name: User specified name. For example: values
 - ii. Config File: User created or as specified by the sample being used
 - iii. Data source type: REST
 - iv. Incoming timestamps: Utc (if using one of the samples).

Header and data format

1. Header format

The REST endpoint supports the HTTP PUT method with the following headers:

```
'content-type': text/html', 'Accept': 'text/plain'
```
2. Data format

Text format that can be parsed by the configured config. file.

Prerequisites to run samples

Each sample specifies required files and in general will include:

- PI Connector for UFL configuration file(s)
- Code files

Value sample

The following sample is based on data collection for an asset with rpm, temperature and vibration sensors. Each record sent to the Connector is formatted as follows:

devicename:sensorname,timestamp (in UTC/GMT),value

For example:

00-00-00-b2-11-1a:rpm,2015-11-06T08:19:18Z,1533

00-00-00-b2-11-1a:temperature,2015-11-06T08:19:18Z,93

00-00-00-b2-11-1a:vibration,2015-11-06T08:19:18Z,2383

Python

Samples were created using Python 3.4.3

Module(s) required

requests

File(s) required

piuflgen.py - generate records

piuflput.py – send a file to the REST endpoint; Requires requests module.

Value.ini – PI Connector configuration file

Procedure

1. Review the setup section and configure a data source using the value.ini file
2. Generate some random data

```
python piuflgen.py value > value.csv
```

3. Send the data file to the PI Connector for UFL REST endpoint

Note: The URL for the REST endpoint is found in the Address field of the Data List Configuration for the configured data source

```
python piuflput.py URL datafile
```

Example

```
Python piuflput.py https://server:5460/connectordata/value value.csv
```

Values sample

The following sample is based on data collection for an asset with rpm, temperature and vibration sensors. Each record sent to the Connector is formatted as follows:

devicename,timestamp (in UTC/GMT),rpm,temperature,vibration

For example:

00-00-00-b2-11-1a,2015-11-06T08:19:18Z,1533,86,2755

Python

Samples were created using Python 3.4.3

Module(s) required

requests

File(s) required

piuflgen.py - generate records

piuflput.py – send a file to the REST endpoint; Requires requests module.

Values.ini – PI Connector configuration file

Procedure

1. Review the setup section and configure a data source using the values.ini file
2. Generate some random data

```
python piuflgen.py values > values.csv
```

3. Send the data file to the PI Connector for UFL REST endpoint

Note: The URL for the REST endpoint is found in the Address field of the Data List Configuration for the configured data source

```
python piuflput.py URL datafile
```

Example

Python piufput.py <https://server:5460/connectordata/values> values.csv

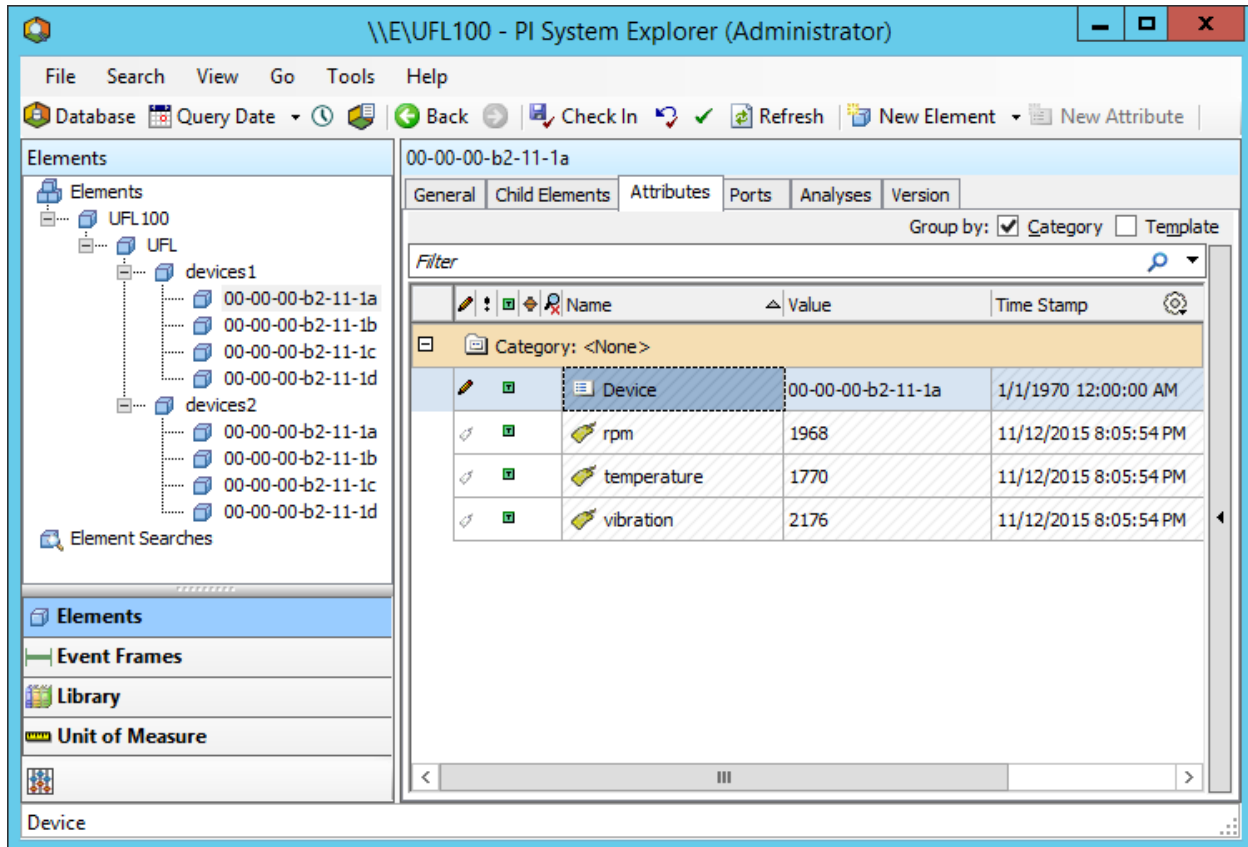


Figure 1 - Example hierarchy created using the value.ini and values.ini configuration files