Coding Applications with the PI Web API

Presented by Ray Verhoeff and Mike Sloves

Agenda

- Assumptions
- Why RESTful vs SOAP
- PI Web API 2015 R3
- Coding Examples
- Demos
- Questions

Our Assumptions

- You know what CODE is!
- You hopefully know at least one programming language that we'll cover.
- You are familiar with the basics of REST principles.
- You'll leave here wanting to play with the PI Web API.
- You'll want to see more code and fewer slides.

RESTful vs SOAP

- Web Services when platform matters!
- SOAP
  - Extremely tight coupling between client and service
  - Not widely and equally supported on all platforms
  - Outdated and deprecated
- REST
  - No client libraries required specifically for the protocol
  - Self describing protocol
  - Widely adopted for cross-platform communication

PI Web API 2015 R3

- A feature-rich collection of RESTful services..
  - AF Create/Read/Update/Delete
    - Elements
    - Attributes
    - Templates
    - PI Points
  - Event Frames Create/Read/Update/Delete
  - AF Search
    - Elements, Attributes, and others
  - PI Point (Time Series) operations
    - Individually or via AF
  - Bulk Read and Write operations

PI Web API 2015 R3

- Multiple Security models
  - Anonymous
  - Basic
  - Kerberos
- CORS Support
- HATEOS
- Self-describing JSON
- Indexed Search
- Live Help

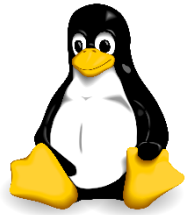
PI Web API 2015 R3 – New CTP Features

- Batch Processing of related and unrelated requests*
- Web Socket (streaming data)*

PI Web API 2015 R3

Only 42 more slides to go…

Supported Platforms



Coding Examples…C#

```
class Program
    {
        private const string URL = @"https://restdemo.osisoft.int/piwebapi/assetdatabases/D0NxzXSxtlKkGzAhZfHOB-
KA0KXjBfjEwUWSsZvL8ridZgUEhMQUZTMDRcQ0hPQyBNSUxL/elements";
        private const string urlparams = "";

        static void Main(string[] args)
        {
            HttpClient client = new HttpClient();
            client.BaseAddress = new Uri(URL);
            client.DefaultRequestHeaders.Accept.Add(
              new System.Net.Http.Headers.MediaTypeWithQualityHeaderValue("application/json"));
            JObject joResult;

            HttpResponseMessage response = client.GetAsync(urlparams).Result;
            if (response.IsSuccessStatusCode)
            {
                string sjson = response.Content.ReadAsStringAsync().Result;
                joResult = JObject.Parse(sjson);
            }
            else
            {
                Console.WriteLine("{0} {1}", (int)response.StatusCode, response.ReasonPhrase);
            }

            Console.ReadLine();
        }
    }
```

Coding Examples…Javascript

```javascript
// Retrieves stream values for an attribute and invokes the callback.
function getStreamValues(attribute, startTime, interval, onStreamValues) {
    var url = attribute.Links.InterpolatedData;
    url += '?startTime=' + startTime + '&';
    url += 'interval=' + interval;
    makeRequest(url, onStreamValues);
}

// Makes an AJAX GET request.
function makeRequest(url, onResponse) {
    var request = new XMLHttpRequest();
    request.onreadystatechange = function() {
        if (request.readyState == 4 && request.status == 200) {
            onResponse(JSON.parse(request.responseText));
        }
    }

    request.open('GET', url, true);
    request.setRequestHeader('Authorization', 'Basic c3R1ZGVudDAxOnN0dWRlbnQ=');
    request.setRequestHeader('Accept', 'application/json');
    request.send();
}
```

Coding Examples…Python

```
import urllib
import urllib2
import base64

url = "https://restdemo.osisoft.int/piwebapi/assetdatabases/D0NxzXSxtlKkGzAhZfHOB-
KA0KXjBfjEwUWSsZvL8ridZgUEhMQUZTMDRcQ0hPQyBNSUxL/elements"
headers = {"Content-Type":"application/json"}

request = urllib2.Request(url)

for key,value in headers.items():
    request.add_header(key,value)

response = urllib2.urlopen(request)

print response.info().headers
print response.read()
```

Coding Examples…C++ (Ray's Favorite)

Casablanca

- http://casablanca.codeplex.com/
- Microsoft open-source project for REST-based communication in native code
- Support for:
  - Windows
  - Linux
  - Mac OS X
  - iOS
  - Android
- NuGet packages for Windows and Android
- Visual Studio 2012, 2013 and 2015

Casablanca

- Features:
  - HTTP(S) Client and Server
  - JSON Serialization
  - Asynchronous streams
  - WebSockets client
  - oAuth Client (both 1 and 2)
  - Parallel Patterns Library (PPL) tasks

PPL Programming

```
http_client client(url);
pplx::task<http_response> reqtask = client.request(methods::GET);

http_response response = reqtask.get();

utility::string_t payload = response.extract_string().get();

Concurrency::task_status stat = reqtask.wait();
```

PPL Programming – when_all

```
std::vector<pplx::task<http_response>> reqtask;

for (int i = 1; i < 10; i++)
{
        http_client client(wUrl[i]);
        reqtask.push_back(client.request(methods::GET));
}

auto list = Concurrency::when_all(reqtask.begin(), reqtask.end());

Concurrency::task_group_status stat = list.wait();
```

Cool Coding Examples…Batch

```
{
    "1": {
        "Method": "GET",
        "Resource": "https://localhost/piwebapi/points?path=\\\\myPIServer\\cdt158",
        "Headers": {
            "Cache-Control": "no-cache"
        }
    },
    "2": {
        "Method": "GET",
        "Resource": "https://localhost/piwebapi/streams/A0EDqD5loBNH0erqeqJodtALAUvtkqIDl4xG-twAFmjx6AA79jxF7HiqFwpsgMBxwgWmgUkVTVFVOSVRcU1VNJQVHQU5cTlVHUkVZTlxIT1VTVE9f0FBMQU5UQ0lUWQ/recorded?starttime=2014-01-01&endtime=2014-01-02",
        "Headers": {
            "Cache-Control": "no-cache"
        }
    },
    "3": {
        "Method": "GET",
        "Resource": "{0}?starttime={1}&endtime={2}",
        "Parameters": ["$.1.Content.Links.RecordedData","$.2.Content.Items[0].Timestamp","$.2.Content.Items[-1:].Timestamp"],
        "ParentIds": ["1","2"]
    }
}
```

Cool Coding Examples…Channels

```
var uri = "wss://myserver/piwebapi/streams/{webId}/channel";

var webSocket = new WebSocket(uri);

webSocket.onopen = function(event)
{
     console.log("Connection opened.");
};

webSocket.onerror = function(event)
{
     console.log("Connection aborted.");
};

webSocket.onclose = function(event)
{
     console.log("Connection closed.");
};

webSocket.onmessage = function(event)
{
     console.log("Message received: " + event.data);
};
```

Contact Information

Ray Verhoeff

ray@osisoft.com

Product Manager

Developer Technologies

Mike Sloves

msloves@osisoft.com

Group Lead

Developer Technologies

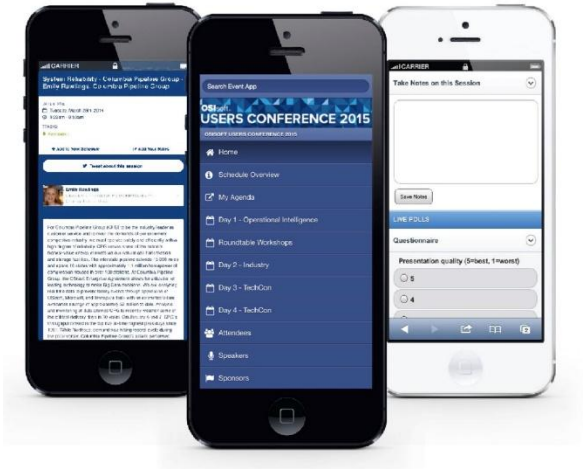
Questions

Please wait for the **microphone** before asking your questions

State your
name & company

Please don't forget to…

Complete the Online Survey for this session



*link not yet provided

감사합니다

谢谢

Danke

Gracias

Merci

Thank You

ありがとう

Спасибо

Obrigado