

Developer Technologies

Which are the best tools and technologies to use for your project

Bodo Bachmann, Frank Garriel, & Justin Starnes

Your Company

IT Devs, Business Analysts, Data Scientists

CHALLENGE

What Developer Technology should I use to access PI data?

- Platform requirements
- Use cases
- Familiarity

SOLUTION

AF SDK, PI Web API, & PI SQL Family

- AF SDK - .NET, Server-side, Performance
- PI Web API – Platform Independence
- PI SQL Family – Reporting & Relational DB Integration

RESULTS

Made a problem-solving application

- Nailed it!

Which Developer Technology to use

	AF SDK	PI Web API	PI SQL Family
Windows + .NET	☐	☐	☐
Platform Agnostic	☐	☐	☐ *
Server-side Apps	☐	☐	☐
Web & Mobile Apps	☐	☐	☐
BI & Data Science	☐	☐	☐
Data Ingress to PI	☐	☐	☐

* Many platforms supported

AF SDK

AF SDK

Most performant data access methodology for your PI data in a Windows environment.

Most flexible data access methodology for getting data in and out of the PI System.

It has full support for handling timestamps with your time-series data (ex. * and *-10h).

It enables users to write service applications, such as the PI Analysis or Notifications service.

It enables users to create custom applications, including data references, for scenarios not available off the shelf.

Custom Data Reference Example

```
// This data reference reads and writes an integer value that is
// persisted in the ConfigString property
[Serializable]
[Guid("c9d5b773-9d26-42c1-9716-d890c0a56914")]
[Description("SampleCS; CSharp Sample Data Reference.")]
public class SampleDR : AFDataReference
{
    int _value = 42;
    public override AFDataReferenceMethod SupportedMethods
    {
        get
        {
            return (AFDataReferenceMethod.GetValue | AFDataReferenceMethod.GetValues | AFDataReferenceMethod.SetValue);
        }
    }
    public override AFDataMethods SupportedDataMethods
    {
        get
        {
            return base.DefaultSupportedDataMethods;
        }
    }
    public override AFValues GetValues(object context, AFTimeRange timeRange, int numberOfValues, AFAttributeList inputAttributes, AFValues[] inputValues)
    {
        return base.GetValues(context, timeRange, numberOfValues, inputAttributes, inputValues);
    }
    public override AFValue GetValue(object context, object timeContext, AFAttributeList inputAttributes, AFValues inputValues)
    {
        return new AFValue(_value, AFTime.Now);
    }

    // In most cases the ConfigString stores information
    // on how to retrieve the data values instead of the
    // data itself, and the data value from a SetValue call
    // is stored in an AFAttribute, PI tag, or other store.
    // In this simple example we store the data itself in the ConfigString.
    public override void SetValue(object context, AFValue newValue)
    {
        if (newValue != null && newValue.Value != null)
            this.ConfigString = newValue.Value.ToString();
        else
            this.ConfigString = String.Empty;
    }
    public override string ConfigString
    {
        get
        {
            return this._value.ToString(CultureInfo.InvariantCulture);
        }
        set
        {
            if (System.Int32.TryParse(value, out _value))
            {
                base.ConfigString = value;
                base.SaveConfigChanges();
            }
        }
    }
}
```

Custom Data Reference Implementation

Implement the basic functionality of getting and setting value.

1. Determine Supported Methods – GetValue, SetValue, etc.
2. Determine Supported Contexts – RecordedValue, InterpolatedValue, etc.
3. Implement methods and account for context, such as time, time range, etc.

Data Pipe Example

```
protected override bool GetEvents()
{
    // Set evaluation time to current time
    ATime evalTime = ATime.Now;

    // Get values for each AF Attribute, one at a time
    foreach (AFAttribute att in base.Signups)
    {
        if (!ReferenceEquals(att, null))
        {
            // Add AF Attribute if it hasn't been added to the _lastTimes dictionary yet
            if (!_lastTimes.ContainsKey(att))
            {
                _lastTimes.Add(att, this._startTime);
            }

            // Set time range to get all values between last timestamps to current evaluation time
            ATimeRange timeRange = new ATimeRange(_lastTimes[att], evalTime);

            /* Note: Use RecordedValues if supported. GetValues call return interpolated values at the start and end time,
             * which can be problematic in a data pipe implementation. GetValues is used here for this simple example because
             * the implementation of GetValues in my custom DR does not return interpolated values at the start and end time. */
            AFValues vals = att.GetValues(timeRange, 0, att.DefaultUOM);

            // Store old last time for the AF Attribute
            ATime lastTime = _lastTimes[att];

            // Publish each value to the data pipe
            foreach (AFValue val in vals)
            {
                // Record latest timestamp
                if (val.Timestamp > lastTime)
                {
                    lastTime = val.Timestamp;
                }

                AFDataPipeEvent ev = new AFDataPipeEvent(AFDataPipeAction.Add, val);
                base.PublishEvent(att, ev);
            }

            // Add a tick to the latest time stamp to prevent the next GetValues call from returning value at the same time
            _lastTimes[att] = lastTime + TimeSpan.FromTicks(1);
        }
    }

    return false;
}
```


Data Pipe Implementation

Implement the AFDataPipe to get notified of data changes in the server.

1. Inherit from AFEventSource

- Store timestamps when data last retrieved for each attribute
- Override GetEvents to publish events between last and current time
- Dispose of timestamps

2. Create Data Pipe

3. Include Data Pipe in supported data methods

This allows a persisted, long running, service to process new data arrival at the PI System to support near real time operations.

Review of Approach

Ease – It only takes a few lines of code to use the AF SDK to access your data.

Flexibility – It allows anything that can be done in code to be used.

Performance/Scalability - AF SDK provides a data cache that can be used to improve performance/scalability.

Security – Be aware of complexity in handling authentication and authorization for multiple users.



➞ Product Booth

- PI Asset Framework
- Developer Technologies

Product Expo, Golden Gate – Hilton



➞ Talks & Labs

High Performance Applications

Today - 10:30

PARC 55: Mission II

Rolling out Asset Framework

Today - 11:30

HILTON: Imperial Ballroom A

Monitor PI Analysis Service

Today - 2:30

PARC 55: Mission II



➞ And more ...

PI Square

<https://pisquare.osisoft.com>

Custom Data Reference
Example

[PI Square Link](#)

Newly Released Features (AF 2018 SP2)

- Improved performance when searching for event frames
 - Overlapped or Active Between
(event time range crosses queried time range)
 - Value across multiple templates and/or non-templated attributes
- Improved flexibility in searches
 - OR operator
 - OwnerName
- Administration functions (ex. clear change tables or re-index)
 - § Previously only available using AFDiag
 - § Now available via AF SDK or PowerShell

* The list above is developer focused and incomplete. *

AF SDK



- Justin Starnes
- Senior Software Developer
- OSIsoft
- Jstarnes@osisoft.com

PI Web API

Frank Garriel

Donkey Kong PI World
2018 Champion

fgarriel@osisoft.com



Frank Garriel

Ms. Pac-Man PI World
2019 Champion

fgarriel@osisoft.com



Frank Garriel

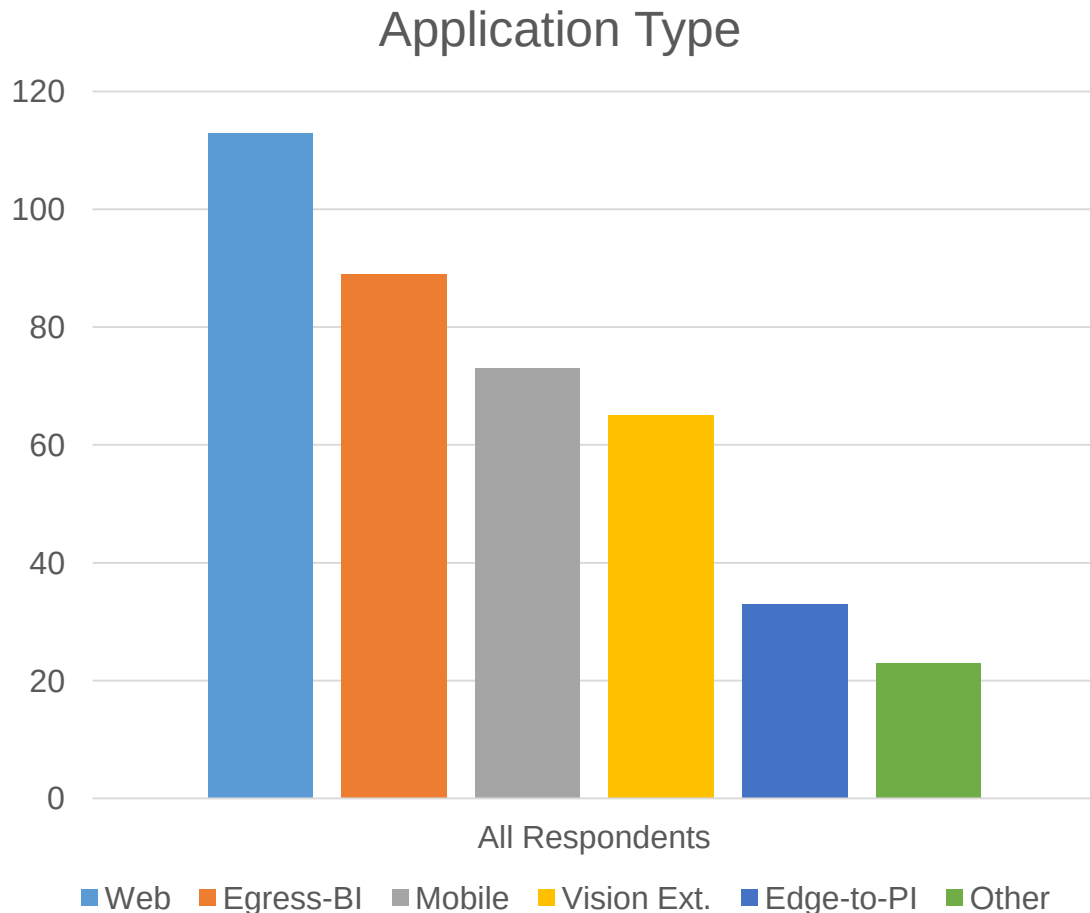
PI Web API PI World
2019 Champion

fgarriel@osisoft.com

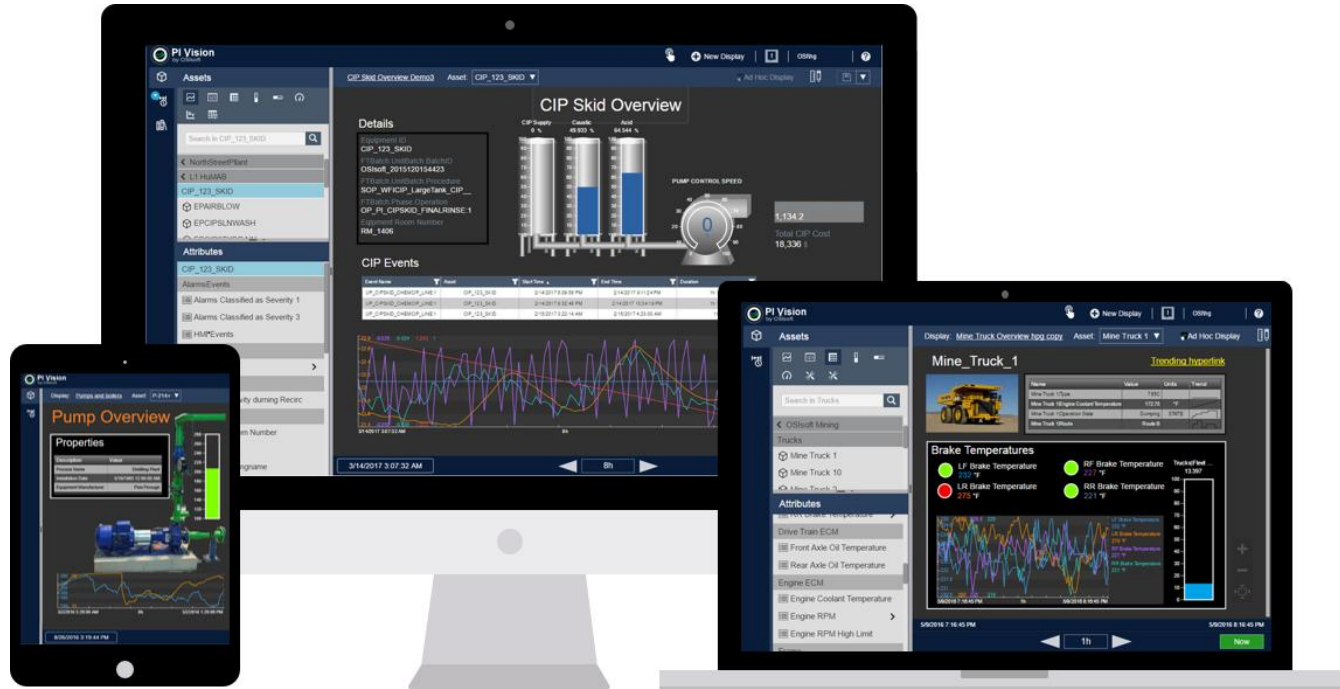


What you make with PI Web API

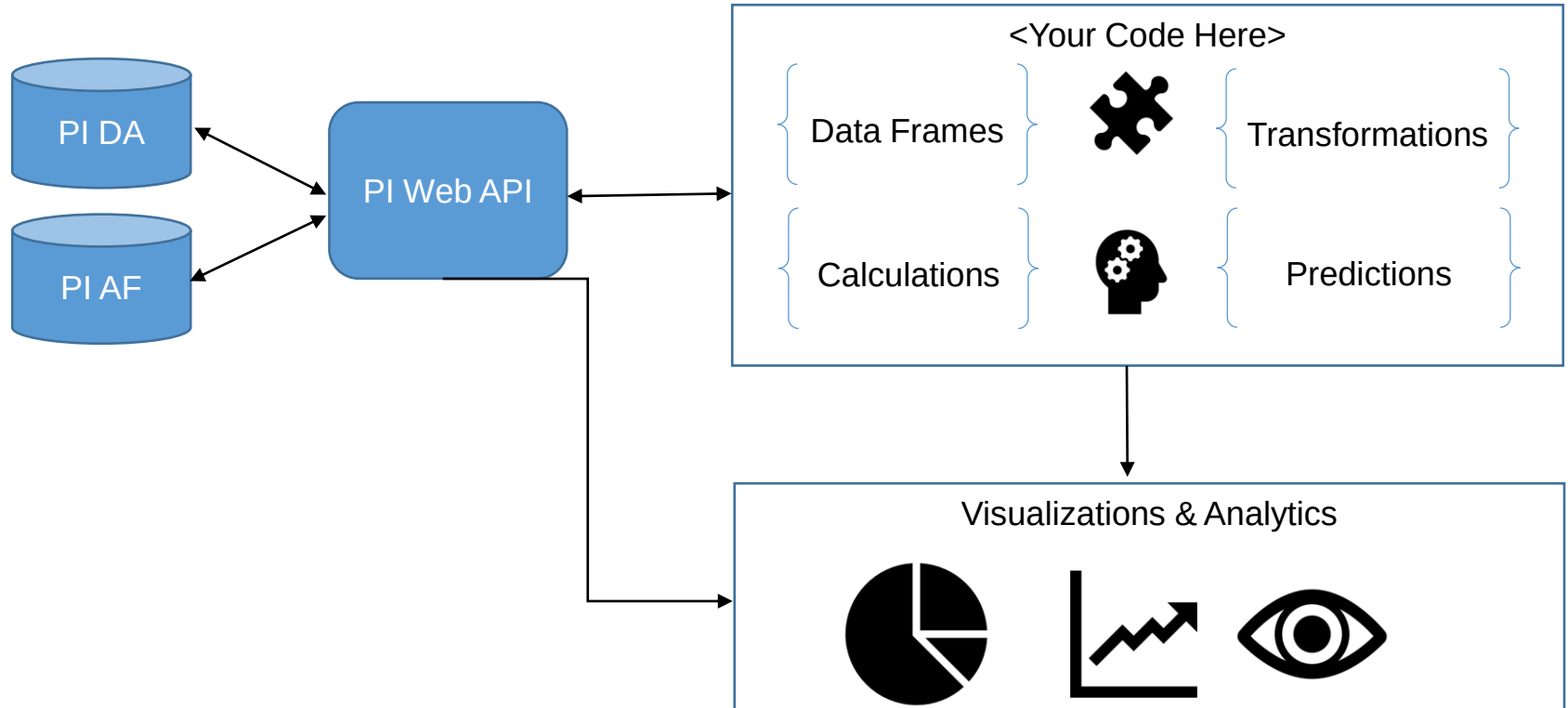
1. Web Apps
2. Data Science & BI
3. Mobile Apps
4. PI Vision Extensibility
5. Edge data to PI



Perfect for Web & Mobile Applications



Great for Data Science



Great for Data Science

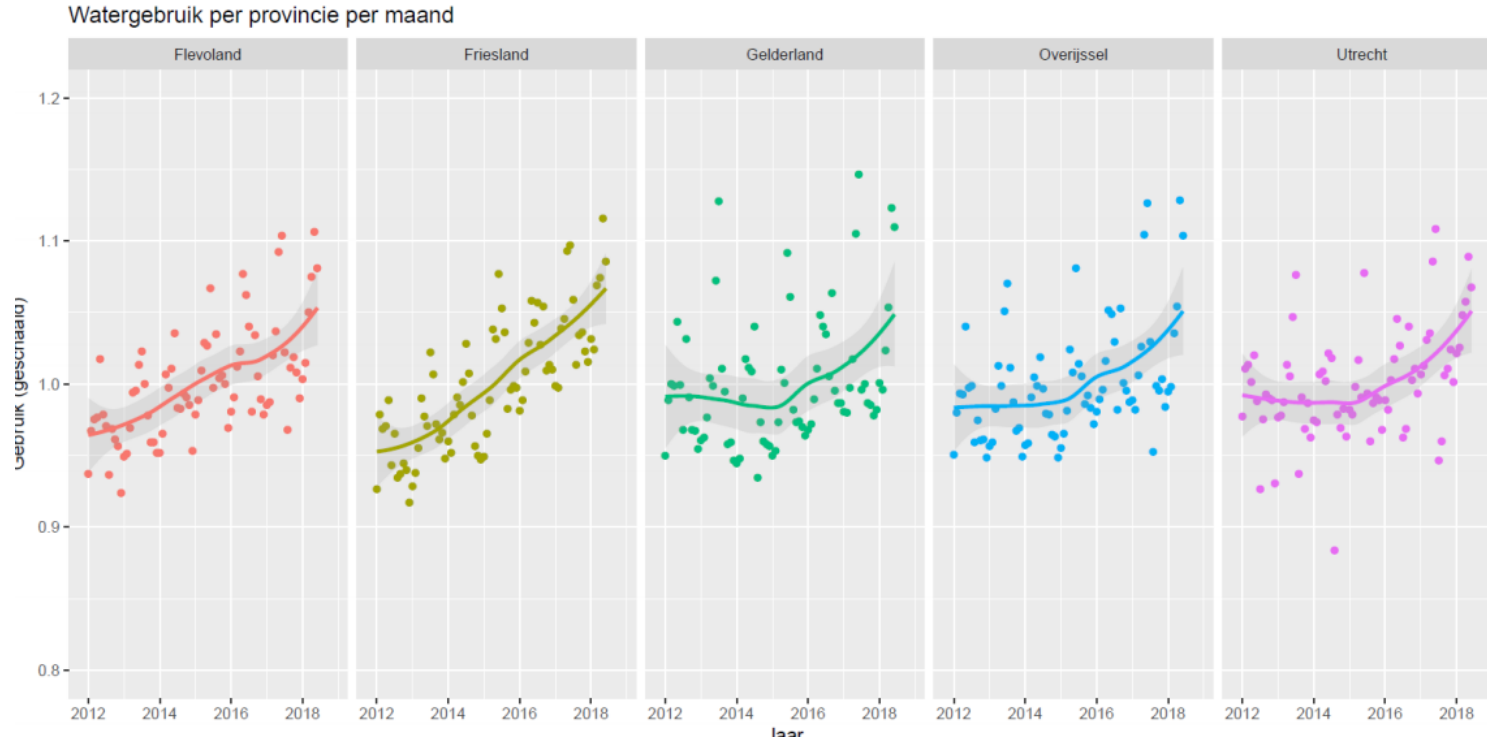
```
library(piwebapi)
```

```
pi = piwebapi$new(pi_server_address, FALSE, username, password, FALSE, TRUE)
```

```
data = pi$data$getInterpolatedValues(path = 'pi:\\\\SR-16635\\FR-MOWE-PT01-meetwaarde',  
                                     startTime = '2018-07-01',  
                                     endTime = '2018-08-01',  
                                     interval = '15m')
```

	timestamp	value	unitsAbbreviation	good	questionable
1	2018-06-30T22:00:00Z	322.4664		1	0
2	2018-06-30T22:15:00Z	331.2576		1	0
3	2018-06-30T22:30:00Z	334.9206		1	0
4	2018-06-30T22:45:00Z	334.7985		1	0
5	2018-06-30T23:00:00Z	337.9731		1	0
6	2018-06-30T23:15:00Z	341.514		1	0

Great for Data Science





➞ Product Booth

- PI Asset Framework
- Developer Technologies

Product Expo, Golden Gate – Hilton



➞ Labs

Building a Manual Data Entry
Symbol In PI Vision
Today, 10:30 AM – 12:30 PM

Getting Started With PI Web API
using Postman
Today, 2:30 PM – 5:30 PM



➞ And more ...

Using PowerApps to Build Real
World Apps With Your PI Data

Generating API clients with Open
API 2.0 (Swagger)

Data Science with R and the PI
System

Available at:
osisoft.com/presentations

Other Available Resources

Tutorials on PI Square -

<https://pisquare.osisoft.com/community/Learn-PI/programming-in-pi-web-api>

Samples on GitHub - <https://github.com/osisoft/PI-Web-API-Samples>

Reference Guide -

<https://techsupport.osisoft.com/Documentation/PI-Web-API/help.html>

What's new and upcoming

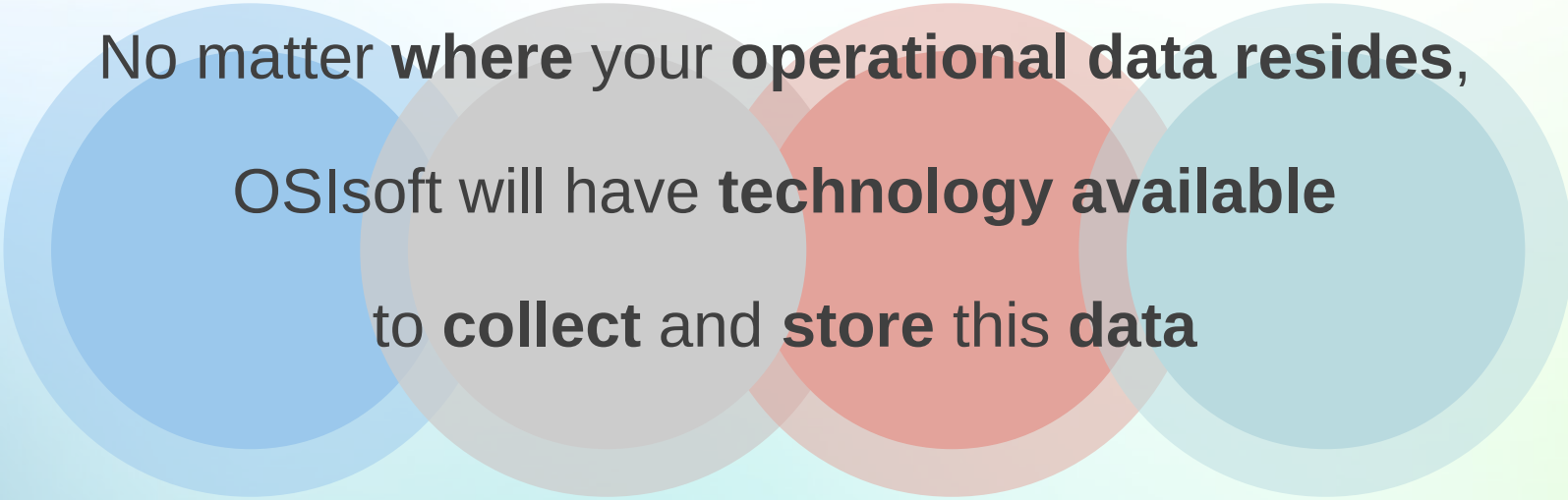
- PI Web API 2018 SP1 (Released?)
 - Stream Updates CTP
 - Increased Streamsets Performance
 - Bug Fixes & Security Improvements

Awesome for Ingress

PI Web API 2019 (Developing Now)

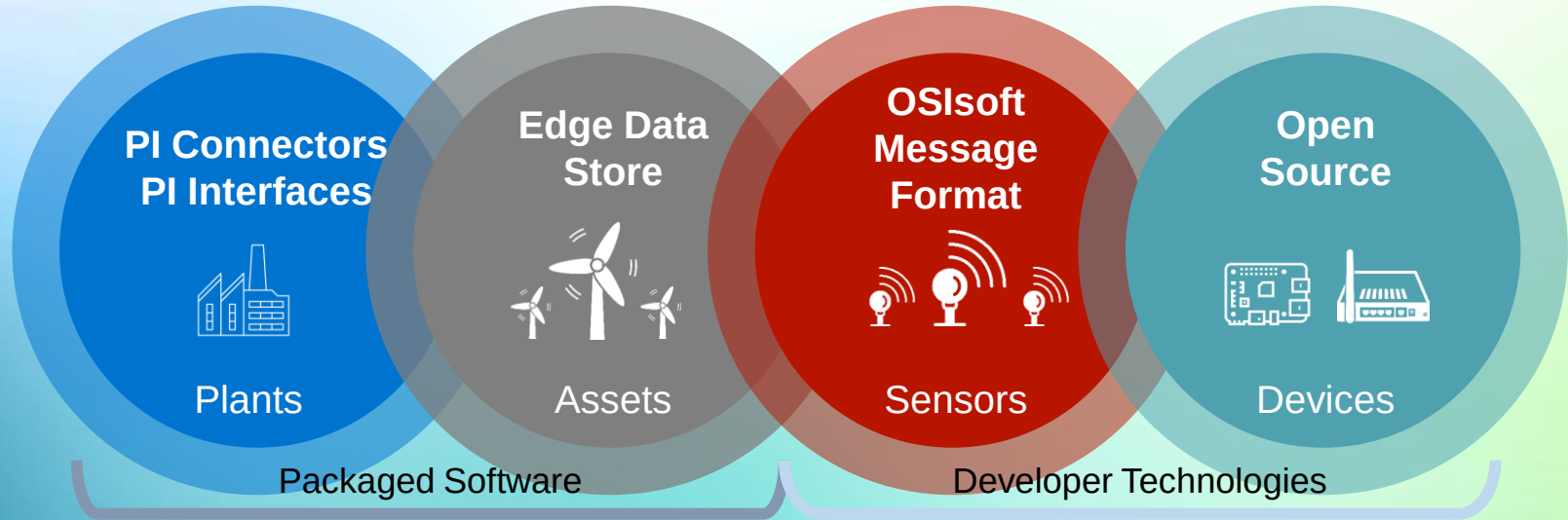
OSIsoft Message Format in PI Web API!

The Goal of Pervasive Data Collection (PDC)

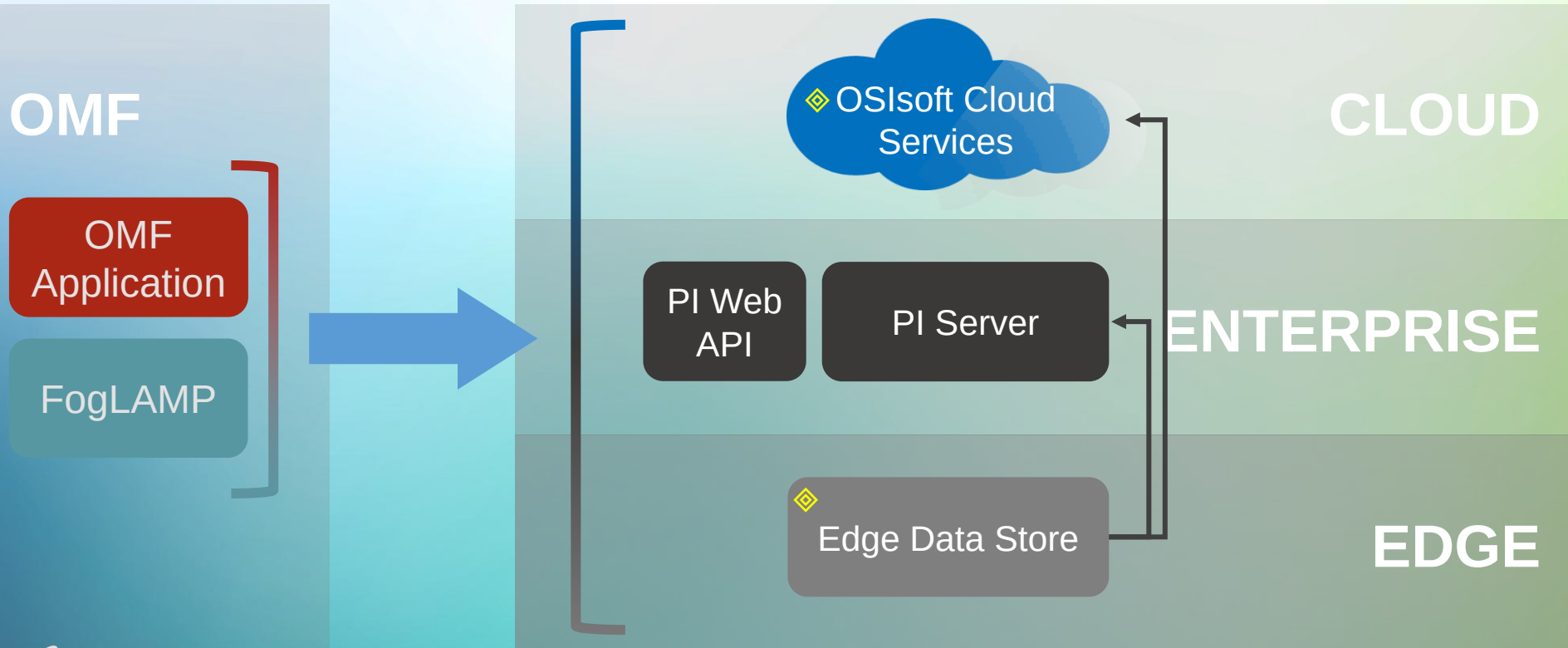


No matter **where** your **operational data** resides,
OSIsoft will have **technology** available
to **collect** and **store** this **data**

Pervasive Data Collection Offerings: OMF and More



Quick Summary of Possible OMF Destinations



Why Use the OSIssoft Message Format?

It allows **writing data** into an **OSIssoft destination** in a manner that is



✓ Flexible



✓ Lightweight



✓ OS, hardware, and language agnostic



✓ Destination agnostic



➞ Product Booth

Developer Technologies
Product Expo, Golden Gate – Hilton

Emerging Technologies
Product Expo, Yosemite – Hilton



➞ Labs

Code your way out of the
stickiest data connectivity
requirements on your next IoT
project

*Today, 10:30-12:30, Lombard,
Parc 55*

Hands-On with FogLAMP:
Getting your IIoT Data into PI
*Today, 2:30-5:30, Fillmore, Parc
55*



➞ And more ...

Designing OMF into A Multi-function
Edge Device (Monico)
osisoftware.com/presentations

What is OMF?
explore.osisoftware.com/omf/

OMF Specification
omf-docs.readthedocs.io/en/v1.1/

//TODO: get better picture of me



Frank Garriel

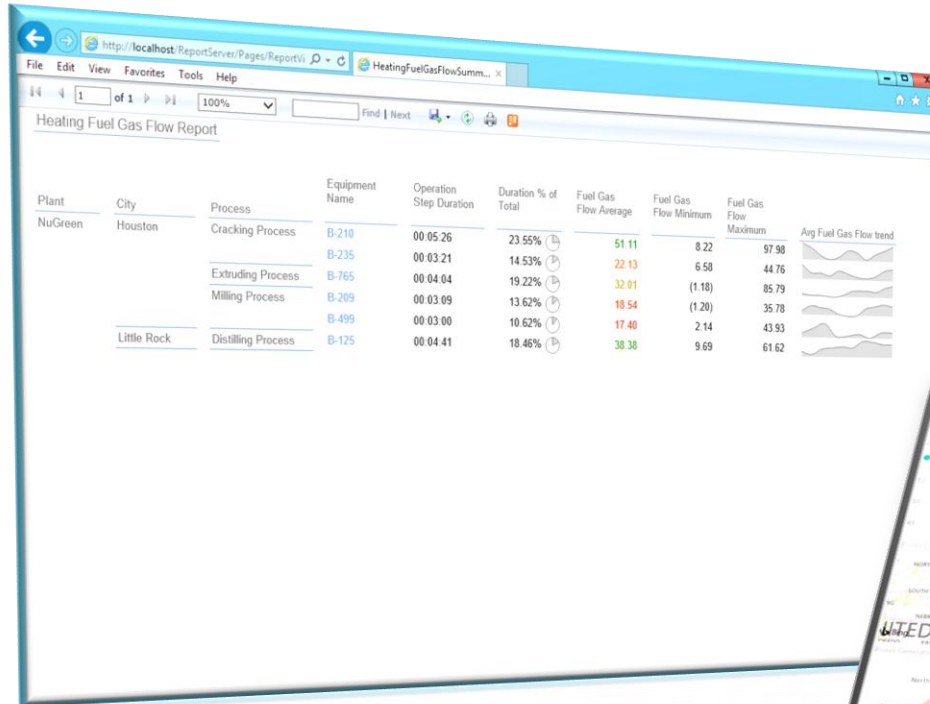
Technical Product Manager, PI Web API
OSIsoft

Fgarriel@osisoft.com

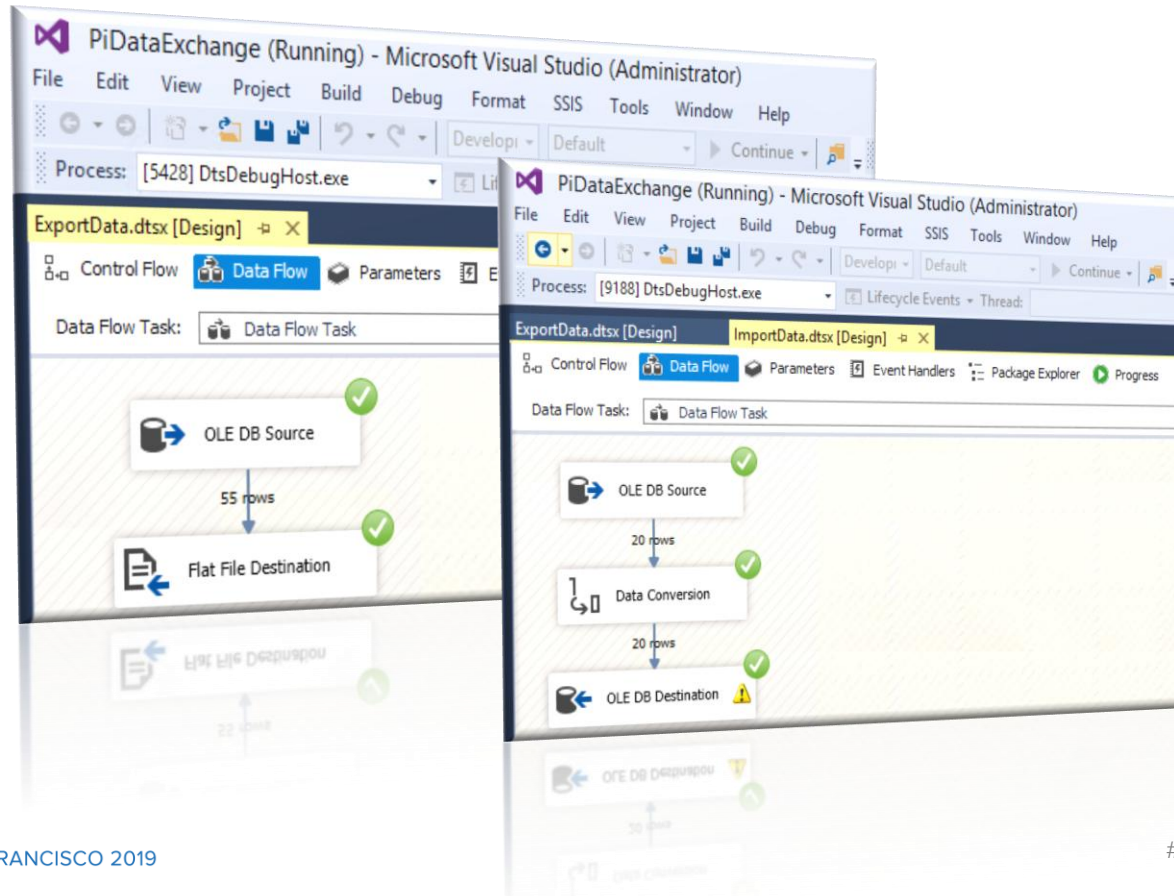
PI SQL Family

Use Cases...

SQL Server Reporting Services



SQL Server Integration Services



SQL Server Linked Server

The screenshot displays the Microsoft SQL Server Enterprise Edition interface. The top menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. The main window is titled 'SQLQuery1.sql - KMITAWPC-AF.master (OS\wkmta (51))' and contains a SQL query. The query is a complex JOIN operation across multiple tables, including Equipment, OperationStep, and Asset. The results are displayed in a table with columns: Equipment Location Path, Equipment Name, Operation Step, Operation Step Start Time, and Operation Step End Time. The status bar at the bottom indicates the query was executed successfully.

Object Explorer

- Connect
- Databases
- Security
- Server Objects
 - Backup Devices
 - Endpoints
 - Linked Servers
 - Providers
 - LINKEDAF
 - Catalogs
 - System Catalogs
 - Configuration
 - NuGreen
 - Tables
 - System Tables
 - Asset.Category
 - Asset.Element
 - Asset.ElementAttribute
 - Asset.ElementAttributeCategory
 - Asset.ElementExtendedProperty
 - Asset.ElementHierarchy
 - Asset.ElementReference
 - Asset.ElementTemplate
 - Asset.ElementTemplateAttribute
 - Asset.ElementTemplateAttributeCategory
 - Asset.ElementTemplateCategory
 - Asset.ElementTemplateExtendedProperty
 - Asset.EnumerationSet
 - Asset.EnumerationValue
 - Asset.ft_GetIPoint
 - Asset.ReferenceType
 - Asset.v.Element
 - Asset.v.ElementAttribute
 - Asset.v.ElementAttributeCategory
 - Asset.v.ElementCategory

SQLQuery1.sql - KMITAWPC-AF.master (OS\wkmta (51))

```

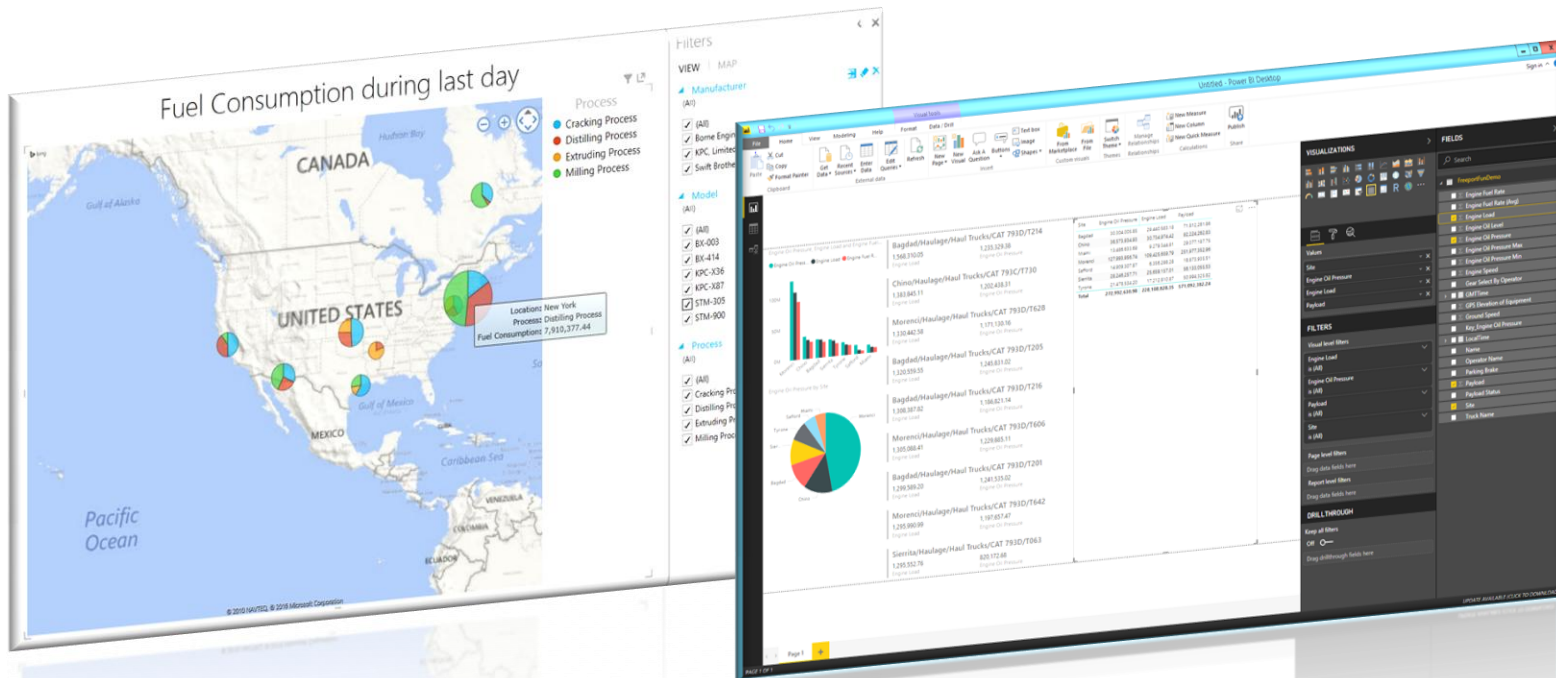
SELECT
    equipment.Path [Equipment Location Path],
    equipment.Name [Equipment Name],
    operationStep.Name [Operation Step],
    operationStep.StartTime [Operation Step Start Time],
    operationStep.EndTime [Operation Step End Time]
FROM [LINKEDAF].[NuGreen].[Asset].[ElementHierarchy] equipment
INNER JOIN [LINKEDAF].[NuGreen].[EventFrame].[EventFrame] operation ON operation.PrimaryReferencedElementID = equipment.ElementID
INNER JOIN [LINKEDAF].[NuGreen].[EventFrame].[EventFrame] operationStep ON operationStep.PrimaryParentID = operation.ID
WHERE operationStep.Name = N'Heating'
ORDER BY [Equipment Name]
  
```

Results

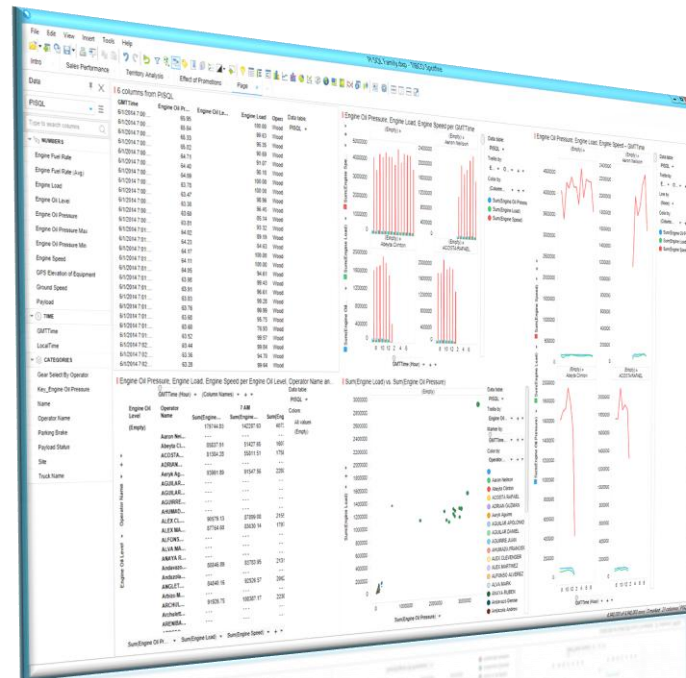
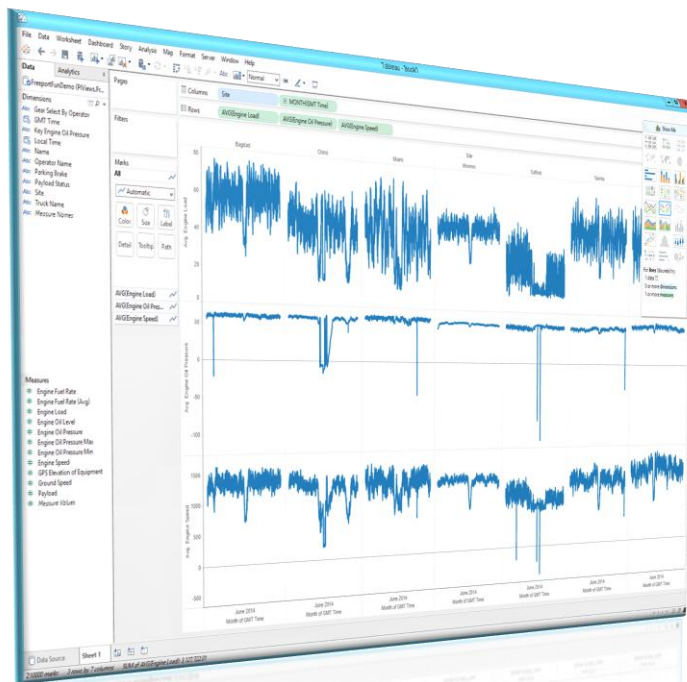
	Equipment Location Path	Equipment Name	Operation Step	Operation Step Start Time	Operation Step End Time
1	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 03:40:21.0000000	2016-01-20 03:42:34.2000000
2	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 05:17:22.5000000	2016-01-20 05:23:33.7500000
3	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 08:14:54.9000000	2016-01-20 08:20:46.3500000
4	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 10:31:10.5000000	2016-01-20 10:35:15.7500000
5	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 12:00:12.0000000	2016-01-20 12:06:48.0000000
6	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 13:04:34.5000000	2016-01-20 13:08:21.7500000
7	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 18:51:20.4000000	2016-01-20 19:00:54.6000000
8	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 20:10:37.5000000	2016-01-20 20:14:56.2500000
9	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 22:36:09.0000000	2016-01-20 22:37:50.0000000
10	\\NuGreen\Little Rock\Distilling Process\Equipment\	B-125	Heating	2016-01-20 22:59:19.5000000	2016-01-20 23:01:12.9000000
11	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 00:31:23.4000000	2016-01-20 00:38:29.1000000
12	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 01:53:46.5000000	2016-01-20 01:56:48.3000000
13	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 04:55:37.5000000	2016-01-20 04:58:26.2500000
14	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 05:50:09.0000000	2016-01-20 05:52:37.5000000
15	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 09:02:27.0000000	2016-01-20 09:05:40.5000000
16	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 10:52:58.5000000	2016-01-20 10:56:14.7000000
17	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 11:53:00.0000000	2016-01-20 11:55:24.0000000
18	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 16:31:09.0000000	2016-01-20 16:33:07.8000000
19	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 17:05:55.9200000	2016-01-20 17:08:06.6000000
20	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 21:34:13.8000000	2016-01-20 21:38:01.5000000
21	\\NuGreen\Houston\Milling Process\Equipment\	B-209	Heating	2016-01-20 22:46:43.5000000	2016-01-20 22:49:05.2500000
22	\\NuGreen\Houston\Milling Process\Equipment\	B-210	Heating	2016-01-20 00:06:37.0000000	2016-01-20 00:08:40.5000000

Query executed successfully. KMITAWPC-AF (120 SP1) OS\wkmta (51) master 00:00:01 64 rows

MS Excel Power View and Power BI

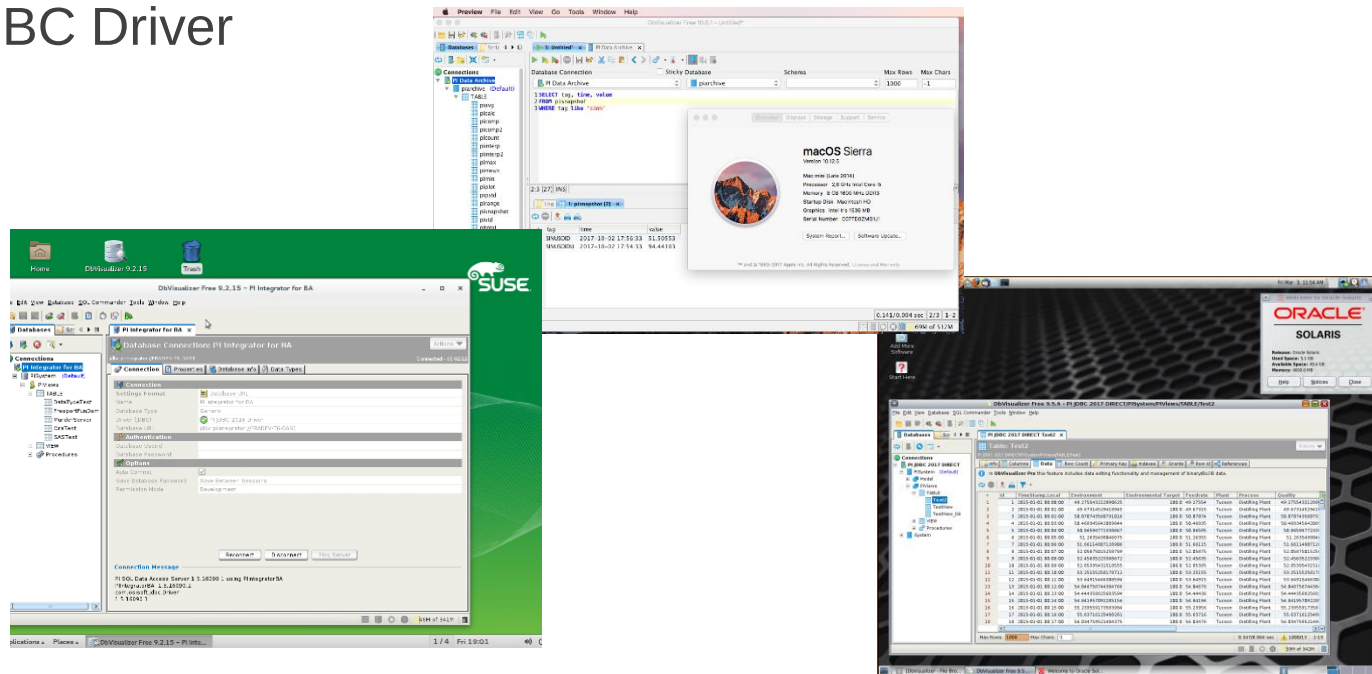


Tibco Spotfire and Tableau



Multi-Platform and Java

- Pure Java JDBC Driver
- Https only



Use Cases for SQL Data Access

- Integrate with 3rd party products (SQL, Office, SAP)
- Create reports (SQL, Crystal, Cognos, etc)
- Share PI data on the Web (SharePoint, ASP.NET, etc)
- Enterprise Application Integration – EAI (BizTalk, Tibco, etc)
- Extract, Transform and Load PI data – ETL (SQL, SAP BO, etc)
- Combine PI with data in other systems (SQL, Oracle, etc)
- Online Analytical Processing – OLAP (SQL)
- Develop custom SQL based applications to PI system
- Manage PI systems
- Support virtually any custom application in any programming language on any platform!

SQL...



PI SQL Commander

PI SQL Query Compendium - PI SQL Commander (Administrator)

FILE EDIT VIEW QUERY PROJECT TOOLS WINDOW HELP

Object Explorer

Query2.sql - KMITAWPC-AF TransposedData.sql - KMITAWPC-AF

***** Object: Custom Function ft_TransposeSummarizeEx_Boiler Script Date: 3/31/2016 7:37:40 PM *****

```

SELECT eh.Path + eh.Name Element, ts.*
FROM [NuGreen].[Asset].[ElementHierarchy] eh
INNER JOIN [NuGreen].[Data].[ft_TransposeSummarizeEx_Boiler] ts
ON ts.ElementID = eh.ElementID
WHERE ts.StartTime = DATE('-1h') AND ts.EndTime = DATE('') AND ts.TimeStep = '30m' AND ts.TimeType = 'MostRecentTime'
OPTION (FORCE ORDER, IGNORE ERRORS, EMBED ERRORS)
    
```

Results

Element	ElementID	StartTime	EndTime	TimeStep	TimeType
1 \NuGreen\Houston\Cracking Process\Equipment\B-235	a90897a9-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
2 \NuGreen\Houston\Cracking Process\Equipment\B-235	a90897a9-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
3 \NuGreen\Houston\Cracking Process\Equipment\B-210	a90897a4-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
4 \NuGreen\Houston\Cracking Process\Equipment\B-210	a90897a4-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
5 \NuGreen\Houston\Cracking Process\Equipment\B-765	a90897db-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
6 \NuGreen\Houston\Cracking Process\Equipment\B-765	a90897db-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
7 \NuGreen\Houston\Cracking Process\Equipment\B-209	a908980b-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
8 \NuGreen\Houston\Cracking Process\Equipment\B-209	a908980b-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
9 \NuGreen\Houston\Cracking Process\Equipment\B-499	a908980e-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
10 \NuGreen\Houston\Cracking Process\Equipment\B-499	a908980e-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
11 \NuGreen\Houston\Cracking Process\Equipment\B-125	a90898a9-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
12 \NuGreen\Houston\Cracking Process\Equipment\B-125	a90898a9-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
13 \NuGreen\Houston\Cracking Process\Equipment\B-045	a9089879-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
14 \NuGreen\Houston\Cracking Process\Equipment\B-045	a9089879-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
15 \NuGreen\Tucson\Cracking Process\Equipment\B-309	a9089781-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
16 \NuGreen\Tucson\Cracking Process\Equipment\B-309	a9089781-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
17 \NuGreen\Tucson\Cracking Process\Equipment\B-555	a9089790-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
18 \NuGreen\Tucson\Cracking Process\Equipment\B-555	a9089790-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
19 \NuGreen\Tucson\Cracking Process\Equipment\B-117	a9089742-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
20 \NuGreen\Tucson\Cracking Process\Equipment\B-117	a9089742-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
21 \NuGreen\Tucson\Cracking Process\Equipment\B-914	a9089760-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
22 \NuGreen\Tucson\Cracking Process\Equipment\B-914	a9089760-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
23 \NuGreen\Tucson\Cracking Process\Equipment\B-737	a908972b-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe
24 \NuGreen\Tucson\Cracking Process\Equipment\B-737	a908972b-bab3-11e5-a973-00155db20000	2016-03-31 18:37:46.000	2016-03-31 19:37:46.000	00:30:00	MostRe

Query executed successfully KMITAWPC-AF (AF Server) 00:00:00.6215 110 rows

DBVisualizer using PI JDB

The screenshot displays the DBVisualizer Free 9.2.15 application window. The title bar indicates the connection is 'PIJdbc/NuGreen/Asset/TABLE/Element'. The interface includes a menu bar (File, Edit, View, Database, SQL Commander, Tools, Window, Help) and a toolbar with various icons. On the left, a 'Connections' tree shows the hierarchy: PIJdbc > 1000Attributes > 1000Attributes2 > Configuration (Default) > debug_EF > ElementVersioning_DB > MH_DB > NuGreen > AF > Asset > TABLE > Element. The main pane shows the 'Table: Element' view with columns: ID, Name, Description, Comment, Revision, HasChildren, HasMultipleVersions, and ElementTemplateID. The table contains 22 rows of data, including entries for Boilers, Pumps, Heaters, and Compressors. The status bar at the bottom shows 'Max Rows: 1000', 'Max Chars: -1', and performance metrics like '0.016/0.015 sec' and '143/17'.

ID	Name	Description	Comment	Revision	HasChildren	HasMultipleVersions	ElementTemplateID
088221a3-d36d-4328-8c90-3ed0f31f2154	B-499	Boiler B-499		1	false	false	9d93e789-6c3d-4350-
41bb085e-3b2a-4e36-a319-7a9a1eb58664	P-778	Pump P-778		1	false	false	b51344c7-e14e-4b7a-
1b4f349a-bf6d-4632-977a-3811b2b17f49	H-044	Heater H-044		1	false	false	d82fcee1-3788-4ab8-
0ae405bb-c969-4b92-90c2-6b6cd108ba1c	K-340	Compressor K-340		1	false	false	87118a54-54da-459f-
6cf0213d-ca18-4fdd-8c81-98c53cbd58fa	F-333	Cooling Fan F-333		1	false	false	4b5b5305-781c-447f-
2b57a100-cdfc-48e8-bcd9-52001076d4f8	H-485	Heater H-485		1	false	false	d82fcee1-3788-4ab8-
b595f913-1568-4081-9c53-4fbdeacd6f5c	F-357	Cooling Fan F-357		1	false	false	4b5b5305-781c-447f-
1ac57c6c-5cda-4278-b7a3-202bbf50e34	Equipment			1	true	false	null
28d1b093-159f-4c65-8e86-49d268f19504	Milling Process	Houston Milling Process		1	true	false	de9ecdf1-c83d-4680-t
c6e0f24e-a806-4b0e-9319-7d6f5b5ee03d	P-209	Cooling Fan P-209		1	false	false	4b5b5305-781c-447f-
c35acfb4-e3b5-4b3b-a360-79e5975de042	P-020	Pump P-020		1	false	false	b51344c7-e14e-4b7a-
efa6367f-a2bf-4837-86d6-38ed489a7b51	B-765	Boiler B-765		1	false	false	9d93e789-6c3d-4350-
5b6a8c45-d01b-41c7-a521-2d880605bd60	F-112	Cooling Fan F-112		1	false	false	4b5b5305-781c-447f-
f6c8a464-22dc-4d95-994b-7b91beb123a3	P-329	Pump P-329		1	false	false	b51344c7-e14e-4b7a-
ee089663-61be-43ae-8ddd-b01bc9c5a01f	K-567	Compressor K-567		1	false	false	87118a54-54da-459f-
ac9203f7-78c5-4c7f-b85e-25745e0c839f	Equipment			1	true	false	null
a9b3264-80c4-440e-a768-9e077ce411c8	Extruding Process	Houston Extruding Process		1	true	false	de9ecdf1-c83d-4680-t
d5fc3956-f0ee-49a3-9282-25fc2d8efc36	P-214	Pump P-214		1	false	false	b51344c7-e14e-4b7a-
e1307f2b-8e96-4bd6-842d-85206f15cc5f	P-560	Pump P-560		1	false	false	b51344c7-e14e-4b7a-
a3da5be3-be7a-48b7-a9d1-66052eed4c6	F-409	Cooling Fan F-409		1	false	false	4b5b5305-781c-447f-
dc4d065-3f07-4a4b-8d1e-b570d841284e	H-230	Heater H-230		1	false	false	d82fcee1-3788-4ab8-
6efda388-edea-427f-ac2b-4b2b7f24332	H-2043	Heater H-2043		1	false	false	d82fcee1-3788-4ab8-

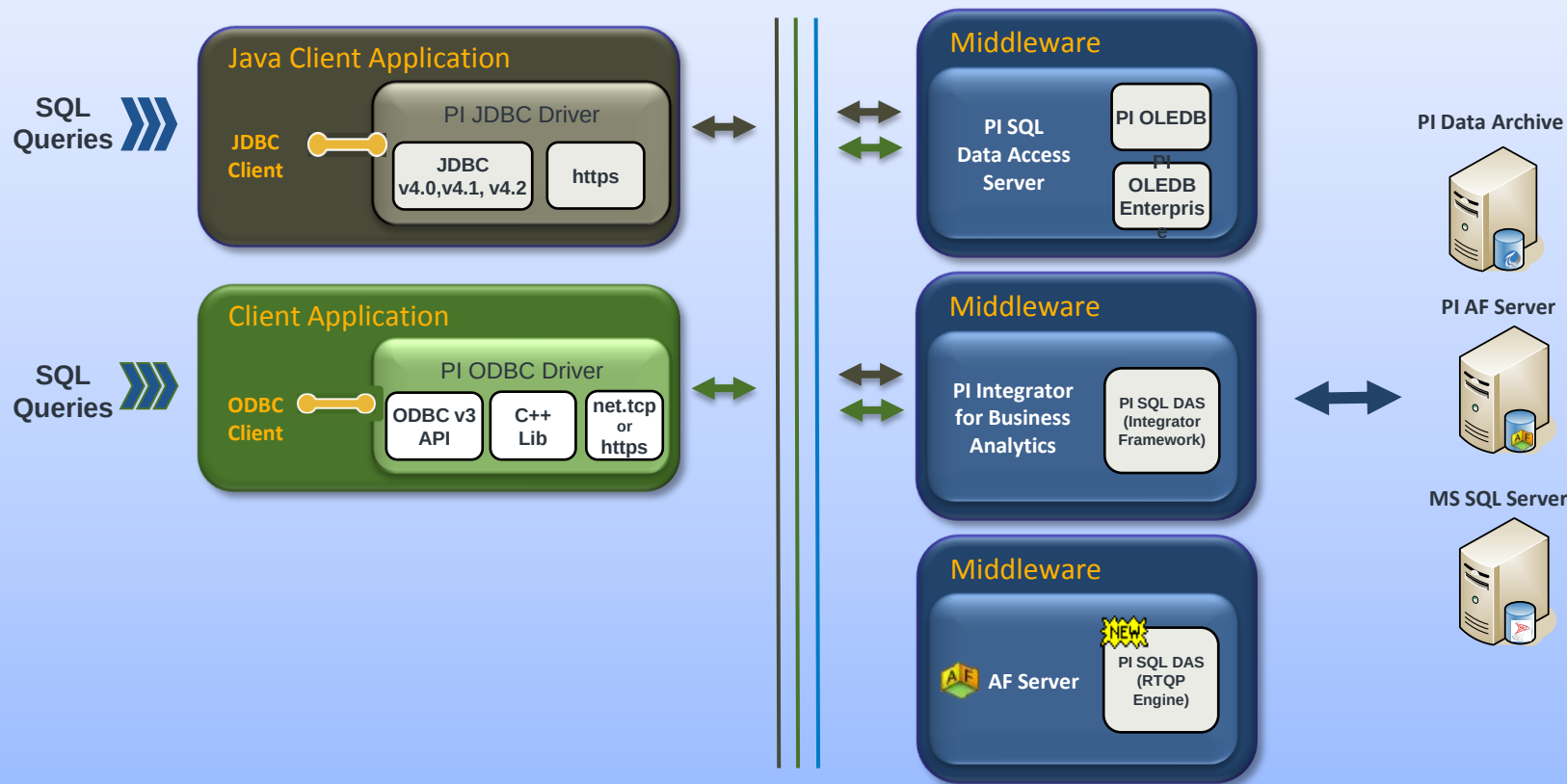
Functional Comparison

Product	Version	PI Point Data Archive	AF Asset	AF Event Frame	AF Attribute Data	Integrator Framework PI Views
PI OLEDB	2017 R2	CRUD	-	-	-	-
PI OLEDB Enterprise	2017 R2	-	R	R	R	-
PI JDBC	2017 R2	CRUD	R	R	R	R
PI ODBC	2016 R2	CRUD	R	R	R	R
PI SQL Client (OLEDB, ODBC, JDBC)	2018 R2	-	R	R	R	-

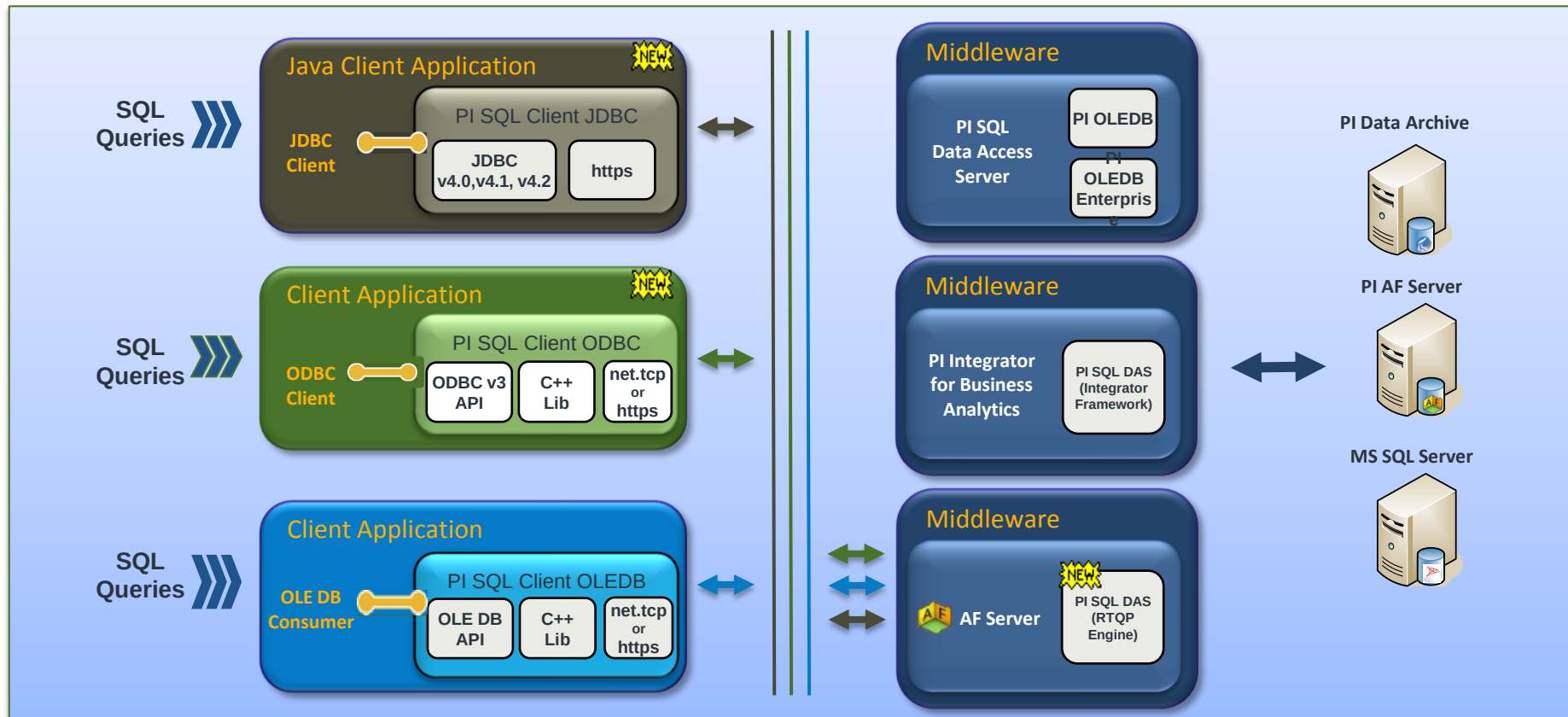
Other functional differences: Aggregation tables, WAN support, Timezone support

Components...

PI JDBC and PI ODBC Architecture



PI SQL Client Architecture



What's coming

PI SQL Client 2018 R2

- Added support for ODBC
- Added support for JDBC (pure Java)

Bug Fix Versions

- PI OLEDB 2019
- PI OLEDB Enterprise 2019

Hands-On Lab and Product Presentation

Lab

- “Mobile Reports: Building Reports For Your Devices Using SQL Server Reporting Services and Power BI”
 - Today, 10:30 AM – 12:30 PM

Tech Talk

- “Modernizing PI SQL for the Future”
 - Today, 2:30 PM – 4:15 PM

PI SQL Family



- **Bodo Bachmann**
- Bodo@osisoft.com
- Engineering Manager Europe
- OSIsoft

ARE YOU READY FOR DAY 3 !

HILTON

- Opening Session & Product Track
- Best Practices Track
- Partner Marketplace Showcase

PARC 55

- Developer Track & Tech Talks
- PI Security Workshop – Not just for Security Gurus!





IT'S THROWBACK THURSDAY AT GEEK NIGHT TONIGHT!

Hilton – Grand Ballroom 7:00 – 10:00 PM

Power Generation User Group Meeting

Parc 55 – Cyril Magnin III
10:30 am – 12:30 pm

Refer to mobile app for exact location and timing

PARTNER & PRODUCT EXPO

Today's Expo Hours

10:00 am – 3:00 pm

Golden Gate and Yosemite



LUNCH IN TWO LOCATIONS TODAY

General Attendee Lunch – Hilton
Developer Lunch – Parc 55





OSIsoft®

PIWorld

Gothenburg • Sept. 16-19, 2019



FINAL TRAINING LABS START TOMORROW AT 9:00 AM

All labs require pre-registration

Check the back of your badge for room locations and visit the registration desk with any questions!

Communicate with OSIsoft Product Managers



<https://feedback.osisoft.com>

.....
If it is not shared on the feedback portal, it didn't happen!

Questions?

Please wait for
the **microphone**

State your
name & company



Please remember

TO DOWNLOAD
APP, SEARCH
OSISOFT



