

*An Introduction to
OPC UA
Open Platform Communication Unified Architecture*

Dan Weise

rev 2b

What is the “window into the process”?

- HMI software or HMI panels are the “window into the process”
 - Operators supervise
 - Process data is displayed and recorded (historian)
 - Video monitors in control rooms and HMI panels everywhere
- It’s how one looks at
 - what is happening (HMI screen)
 - What has happened (historian)
- Integrators do the work of
 - Creating HMI screens
 - Populating HMI screens with specific data
 - Graphics, trend data, pushbuttons/switches, digital panel meters
 - Captive (in-house) or contract (independent)



What is involved in looking at the process?

- We are not integrators, so the tasks involved in getting an HMI package to display or log process data are not obvious.

What's involved in making an HMI?

- Build a tag data base
 - Get the devices to talk HMI software
 - Configure an historian to log the data
 - Develop screens and populate the screens
-
- The three in the red circle will involve the forthcoming OPC UA

HMI Tag Database

Alarm Information 7:53:31 PM 8/18/2016

Current status					
Pen	Pen Name	Reading	Alarm	Type	Set Point
1	Server Room temp	71.33179	P1 Alm 1	High	77
1	Server Room temp	71.33179	P1 Alm 2	High	78
1	Server Room temp	71.33179	P1 Alm 3	High	79
1	Server Room temp	71.33179	P1 Alm 4	High	80
1	Server Room temp	71.33179	P1 Alm 5	High	81
1	Server Room temp	71.33179	P1 Alm 6	High	82

- Building the tag data base
 - Spreadsheet of all the points, includes
 - Tag name
 - Data format (boolean, signed/unsigned integer, Real)
 - Engineering units
 - Scaling factor
 - Scale used in historian
 - Associated properties like alarm values, alarm status, PID values, auto/man
 - Can take days to weeks to build the tag database by hand
 - Some PLC/PAC devices export tag data to a spreadsheet .csv file
 - Spreadsheet has to be re-worked to be compatible with the HMI

HMI Tag Database

Tag Database

Tags to show in the Editor

Show All

☐ Discrete Inputs
☐ Analog Inputs
☒ Integers
☐ System Data
☒ Booleans

Invert

☐ Discrete Outputs
☐ Analog Outputs
☐ Floats
☐ Strings

Editor

Name	Type	Rows	Cols	Retentive	Init Value	Mod Start	Default ...	In Use	Comment
barr750WReadTimeout	Boolean, 1D Array	1	10	<input type="checkbox"/>				<input type="checkbox"/>	
narrStatusRegisters	Integer, 16 Bit, 1D...	1	6	<input type="checkbox"/>			Decimal	<input checked="" type="checkbox"/>	
narrHmi200WPositions	Integer, 16 Bit, 1D...	1	8	<input type="checkbox"/>		400021	Decimal	<input checked="" type="checkbox"/>	
System ID									
		Row	Col	Init Value		Mod Start			
AR1516-000003(1)		1		1		0			400021
AR1516-000003(2)		1		2		500			400022
AR1516-000003(3)		1		3		-100			400023
AR1516-000003(4)		1		4		1			400024
AR1516-000003(5)		1		5		200			400025
AR1516-000003(6)		1		6		2			400026
AR1516-000003(7)		1		7		-300			400027
AR1516-000003(8)		1		8		3			400028
narrHmi200WSpeeds	Integer, 16 Bit, 1D...	1	4	<input type="checkbox"/>		400029	Decimal	<input checked="" type="checkbox"/>	
narr750WStatusRegisters	Integer, 16 Bit, 1D...	1	5	<input type="checkbox"/>			Decimal	<input checked="" type="checkbox"/>	
byarrHomingModeSegments	Integer, 8 Bit Unsi...	1	4	<input type="checkbox"/>			Hex	<input type="checkbox"/>	
byarrStepperResponse	Integer, 8 Bit Unsi...	1	10	<input type="checkbox"/>			Hex	<input type="checkbox"/>	
bcdHomingMode	Integer, 16 Bit BCD			<input type="checkbox"/>	0			<input checked="" type="checkbox"/>	
bWriteComplete	Boolean			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
bWriteError	Boolean			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
bWriteInProgress	Boolean			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
bWriteSuccess	Boolean			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
bWriteTimeout	Boolean			<input type="checkbox"/>				<input checked="" type="checkbox"/>	
bReadComplete	Boolean			<input type="checkbox"/>				<input type="checkbox"/>	
bReadError	Boolean			<input type="checkbox"/>				<input type="checkbox"/>	
bReadInProgress	Boolean			<input type="checkbox"/>				<input type="checkbox"/>	

Add Tags

Delete Tags

HMI Tag Database

- What do you start with?
- Example: UDC 3200 Modbus

A.3 Loop Value Integer Register Map

The following table applies to the following instruments: UDC2300, UDC2500, UDC3200, UDC3300, UDC3500, DR4300 and DR4500. This table applies to Loops 1-24 except Loops 2-24 use the addresses shown in Table A-1

Address (hex)	Register (decimal)	Parameter Name	Access	Notes
0000	40001	PV	R	Signed 16 bit integer Prescale * 10 <i>Note 5</i>
0001	40002	RV; Remote Set Point; SP2	R	Signed 16 bit integer Prescale * 10
0002	40003	Working Set Point	R/W	Signed 16 bit integer Prescale * 10 <i>Note 5</i> On a write the instrument will update the proper set point according to the loop's currently selected set point.
0003	40004	Output	R/W	Signed 16 bit integer Prescale * 10 <i>Note 11</i>

- Data in each cell has to be manually entered into each database cell;
 - no export. Very tedious. *Grunt work. Not a glamour task.*

HMI Communications

- Get devices to talk to the HMI software
 - Hardware links (the mules that carry the data)
 - Ethernet
 - RS-485
 - Protocols (the rules – who talks when)
 - Ethernet/IP
 - Profibus
 - Modbus
 - Proprietary
 - Software drivers do this task (like the Fedex logistics manager)



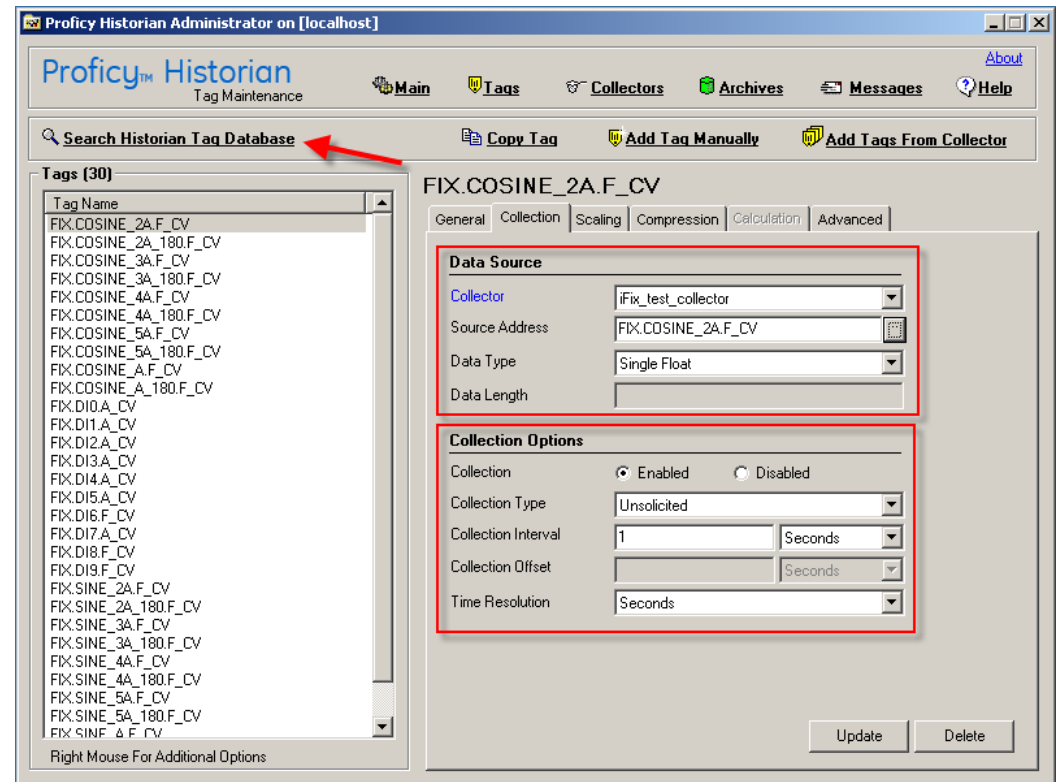
HMI Historian

Historian needs to know

- Which points to log
- How often to log data
- What number format it has to deal with
- Presenting a value – XXX, or XXX.X or XXX.XX or XXX

• Note

- Tag list on the left
- Properties of the tag in the red boxes



Tasks in HMI data display and logging

We've seen some of the effort involved in making an HMI tag and historian database

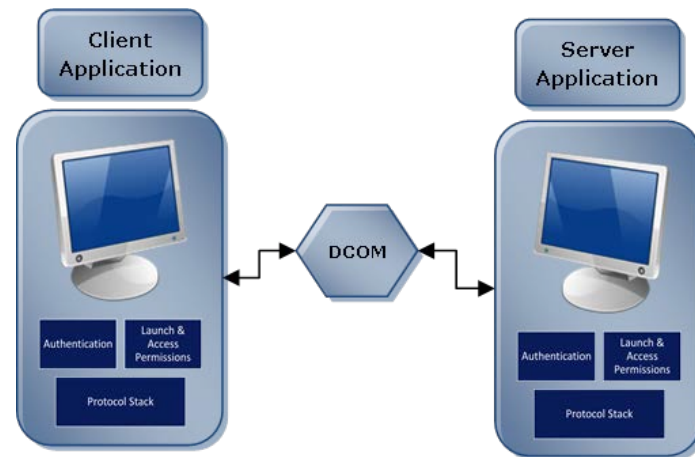
- Build a tag data base
- Get the devices to talk to HMI software
- Configure an historian to log the data



- Develop screens and populate the screens – skip for this discussion
- The one in the red circle involves current OPC DA
- The one in the blue circle involves forthcoming OPC UA

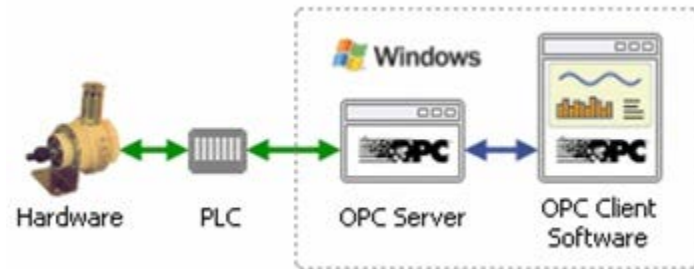
What is OPC ?

- OPC DA has been around since mid 1990's (~20 years)
- OPC DA (Data Acquisition)
 - Very wide spread useage in process controls
- Primarily used by systems integrators
 - Runs in the background, not obvious
- Useable only in Windows products (not Linux)
 - based on DCOM, a Microsoft component
- OPC is all about *data exchange*
 - Read data
 - Write data



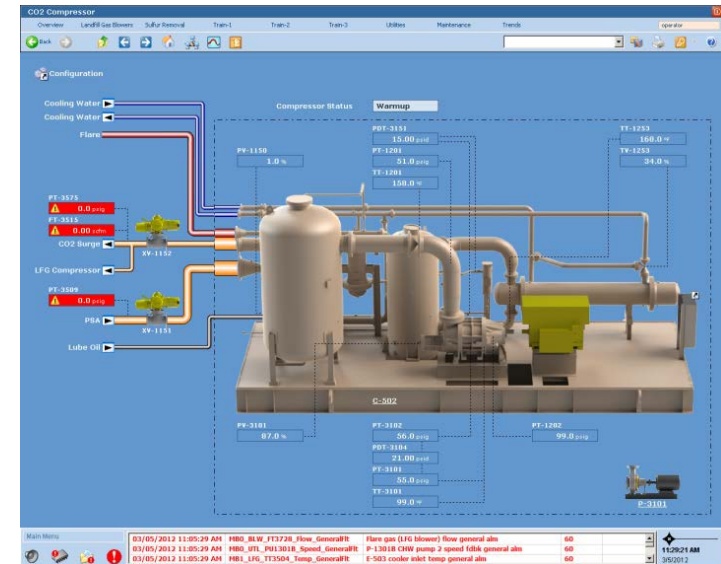
What is an OPC Server?

- OPC servers are ‘hidden’
 - they run in the background
- OPC server talks directly to digital ‘field devices’
- What kind of Field devices?
 - PLCs
 - PACs (HC-900)
 - HMI panels
 - Things that talk digital and need a ‘driver’, like a Modbus driver
 - UDA analyzer, UDC controller (talk Modbus)
 - Trendview
 - Siemens Coriolis FC410 (talks Modbus)
 - Does not include 4-20mA analog field instruments
 - Those need to connect to a system that talks ‘digital’



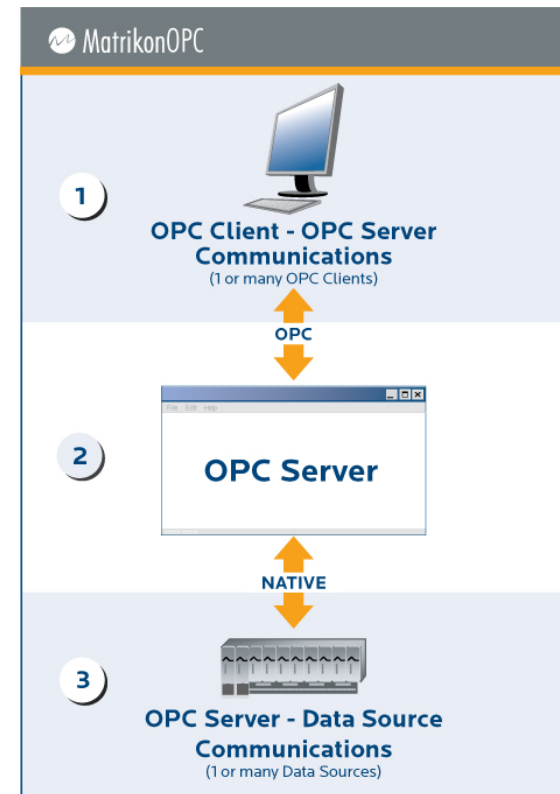
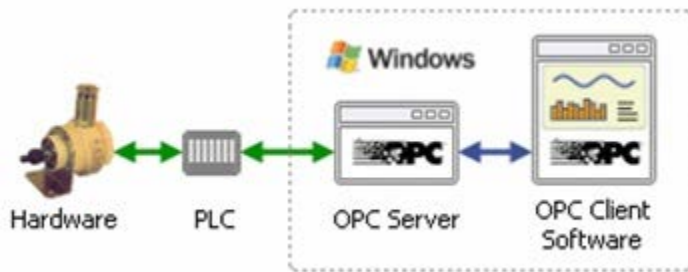
What is an OPC Client?

- OPC client is part of an HMI package
- HMI packages like
 - Wonderware
 - Iconics
 - Citect
 - RSView/FactoryView (Allen Bradley)
 - Win CC (Siemens)
 - Whatever GE calls Intellution nowadays
- When I hear ‘OPC Client’, I think HMI software



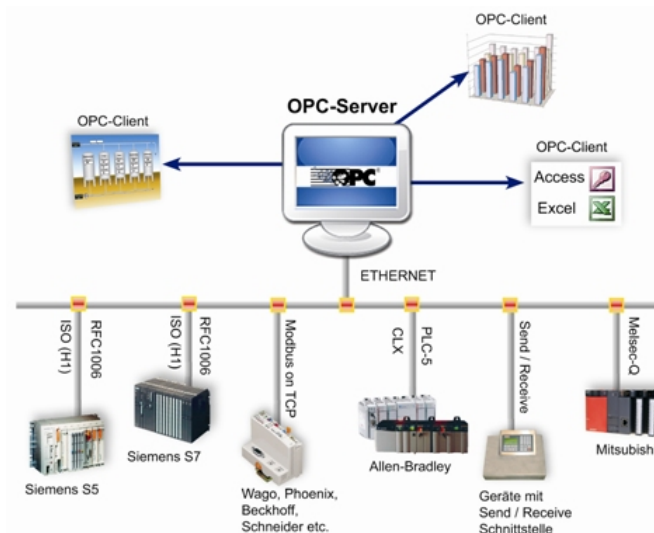
What is OPC DA?

- OPC DA is OPC servers talking to field devices and OPC clients
 - OPC server talks to field devices
 - OPC servers talk to OPC clients
 - OPC client talks to the HMI software



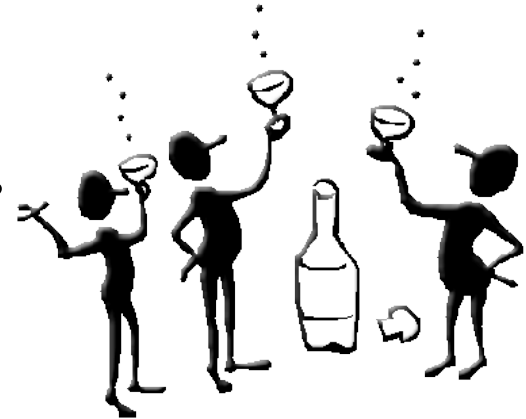
Rules to remember

- Rules
 - Any OPC server talks to any OPC client
 - OPC servers can talk to multiple devices and multiple OPC clients
 - OPC server needs a ‘driver’ for whatever field device it talks to
 - For instance, Kepware OPC server has an HC-900 driver that talks Modbus.
 - An OPC server can run multiple drivers



Why was OPC DA so successful?

- It made the task of getting data from field devices into an HMI much easier
- OPC server vendors specialize in drivers
 - Keep them updated
 - Their customer base is large enough to make writing drivers profitable
- OPC client vendors specialize in aspects of their HMI
 - do not have to bother with drivers
 - Division of labor pays off
- OPC clients talk to OPC servers
- Integrator is involved in configuring both OPC server and client
- Because OPC runs in the background, the OPC Foundation estimates that only 25% of end-users even know that OPC is involved in their HMI.



What is OPC UA?



- OPC UA is the Next Generation OPC
- UA is not dependent on Microsoft's DCOM component
 - DCOM had its issues, configuration, security, robustness
 - Allows Linux based Honeywell One Wireless WDM gateway to use OPC UA
- UA exploits XML, a text based format for data exchange
 - Browsers, like FireFox, Chrome or Internet Explorer can read and display XML
- UA extends OPC to mobile devices, cell phones and tablets
- UA automates most of the effort in building a tag database
 - Recognizes and uses the properties of the tag, not just the value of the tag

Attribute	Value
[-] NodeId	NodeId
[-] Namespa...	2
[-] Identifier...	String
[-] Identifier	TI101.CH01_AI.PV
[-] NodeClass	Variable
[-] BrowseName	2, "PV"
[-] DisplayName	"" , "pv"
[-] Description	"" , "Process Value"
[-] WriteMask	0
[-] UserWriteMask	0
[-] Value	23.4706
[-] DataType	Float
[-] Namespa...	0
[-] Identifier...	Numeric

OPC UA – tags with properties

- Remember all those things associated with a tag?
 - Tag name
 - Engineering units
 - Scaling factor
 - Scale used in historian
 - Report method – time based or report- o- exception
 - Associated elements like alarm values, alarm status, PID, auto/man
- They're all 'properties'

The screenshot shows the 'Tag Database Editor' window. It has a top section for filtering tags to show in the editor, with checkboxes for Discrete Inputs, Analog Inputs, Integers, System Data, Booleans, Discrete Outputs, Analog Outputs, Floats, Strings, and a text input field. Below this is the 'Editor' section, which contains a table of tags. The table has columns for Name, Type, Rows, Cols, Retentive, Init Value, Mod Start, Default, In Use, and Comment. The table lists various tags, including 'barr750WReadTimeout', 'namStatusRegisters', 'namHmi200WPositions', 'System ID', 'AR1516-000003(1)' through 'AR1516-000003(8)', 'namHmi200WSpeeds', 'nam750WStatusRegisters', 'byarrHomingModeSegments', 'byarrStepperResponse', 'bcdHomingMode', 'bWriteComplete', 'bWriteError', 'bWriteInProcess', 'bWriteSuccess', 'bWriteTimeout', 'bReadComplete', 'bReadError', and 'bReadInProcess'. Each tag has its properties filled in, such as type, rows, cols, and initial values.

Name	Type	Rows	Cols	Retentive	Init Value	Mod Start	Default	In Use	Comment
barr750WReadTimeout	Boolean, 1D Array	1	10						
namStatusRegisters	Integer, 16 Bit, 1D...	1	6						
namHmi200WPositions	Integer, 16 Bit, 1D...	1	8			400021	Decimal		
System ID									
AR1516-000003(1)		1			1		0		400021
AR1516-000003(2)		1			2		500		400022
AR1516-000003(3)		1			3		-100		400023
AR1516-000003(4)		1			4		1		400024
AR1516-000003(5)		1			5		200		400025
AR1516-000003(6)		1			6		2		400026
AR1516-000003(7)		1			7		-300		400027
AR1516-000003(8)		1			8		3		400028
namHmi200WSpeeds	Integer, 16 Bit, 1D...	1	4						
nam750WStatusRegisters	Integer, 16 Bit, 1D...	1	5						
byarrHomingModeSegments	Integer, 8 Bit Unsi...	1	4						
byarrStepperResponse	Integer, 8 Bit Unsi...	1	10						
bcdHomingMode	Integer, 16 Bit BCD								
bWriteComplete	Boolean								
bWriteError	Boolean								
bWriteInProcess	Boolean								
bWriteSuccess	Boolean								
bWriteTimeout	Boolean								
bReadComplete	Boolean								
bReadError	Boolean								
bReadInProcess	Boolean								

What is OPC UA?

- For OPC DA, in some cases
 - all data/info had to be entered by hand
 - Many remaining properties had to be entered by hand
- OPC UA grabs all the properties along with the value
 - A tag has ‘properties’, not just a value
 - OPC server gathers the properties
 - OPC server transfers all the associated properties to the client
- Instead of manually entering any data for a database, the database will *‘auto-populate’* with all the tags and their associated properties
 - OPC UA: OPC server will pour the tag database into the the OPC client
- This auto-populate function is well known within a vendor’s line
 - Migrating one from Plantscape to Experion PKS
- OPC UA promises auto-populate across vendor lines



Example 1 – what OPC UA might fix

- Display resolution is configured for an HMI
 - temperature values as whole numbers
 - Pressure values to 3 digits after the decimal point

Menu 0 1y+ 127 Screen		Refr/Freezers Temps 15 Mar 16 09:16:30	
Walk In Cooler 1		42 °F	
Reach In Refrig. 4B	38 °F	Reach In Refrig 4A	38 °F
Walk In Freezer 1	-9 °F	Lab Refrig	41 °F
Micro Refrigerator	41 °F	Walk In Cooler 2	40 °F
Reach In Refrig 2	39 °F	Walk-In Cooler 3	44 °F

Menu 0 1y+ 119 Screen			Room Pressures 15 Mar 16 09:14:10		
Cmpd Rm DP			0.131 in H2O		
Chg Rm DP	0.060 in H2O	Pass In 1 DP	0.060 in H2O	Pass In 2 DP	0.052 in H2O
Pass In 3 DP	0.059 in H2O	Pass In 4 DP	0.087 in H2O	Pass In 5 DP	0.094 in H2O
Pass In 6 DP	0.072 in H2O	Pass Out DP	0.060 in H2O	Staging DP	0.005 in H2O
Cmpd to Chg	0.071 in H2O	Cmpd to Pass Out	0.071 in H2O	Cmpd to P IN #1	0.070 in H2O
Cmpd to P IN #2	0.078 in H2O	Cmpd to P IN #3	0.071 in H2O	Cmpd to P IN #4	0.044 in H2O
Cmpd to P IN #5	0.037 in H2O	Cmpd to P IN #6	0.059 in H2O	Cmpd to Staging	0.126 in H2O

Example 1 – what OPC UA might fix

- Same values displayed in a browser are raw floating point values

Pen Name	Reading	Units
Chg Rm DP	6.765264E-02	in H2O
Cmpd Rm DP	0.1295388	in H2O
Pass In 1 DP	8.973444E-02	in H2O
Pass In 2 DP	2.520309E-02	in H2O
Pass In 3 DP	5.660284E-02	in H2O
Pass In 4 DP	0.1096735	in H2O
Pass In 5 DP	8.999324E-02	in H2O
Pass In 6 DP	6.894965E-02	in H2O
Pass Out DP	6.913279E-02	in H2O
Comp Rm rH	34.87847	% RH
Staging DP	7.882208E-03	in H2O
Label & Pack	63.10789	°F
Fin Prod Vault N	74.11005	°F
Fin Prod vault S	71.25351	°F
Lab Room	74.72298	°F
Lab Refrig	39.83991	°F

- On the recorder screen the value is 0.067, in IE the same value appears as 6.76526E-02
- On the recorder screen the value is 63, in IE the same value appears as 63.10789
 - Exponential format (how convenient)
 - Resolution to 5 or 6 digits after the decimal point (it's mostly noise)

Example 1 – what OPC UA might fix

- Display resolution is a property, has already been defined:

The screenshot shows a configuration window with a tabbed interface. The tabs are 'Edit Setup', 'Pens', 'Pen 49', 'Scale', and 'Numb F...'. The 'Scale' tab is selected. Below the tabs, there are three rows of settings, each with a label on the left and a value in a text box on the right:

Notation	Normal
Auto	User Defined
After Decimal	3

- But IE does not access that property; IE accesses only the value and the engineering units
- Browser has no clue what the Trendview display resolution is
- OPC UA will grab all the object properties
- OPC UA will display the value in any venue the same manner (at the same display resolution) as defined in the source

Example 2 – what OPC UA might fix

- Selecting which tags or points appear on an HMI screen

Edit Layout	Screens	Screen 10
Name	6 valid DPMs	
Enabled	✓	
Template Type	DPMs	
Select By	Pen	
Showing (Pens)	1, 2, 3, 4, 5, 6	

- Although the pens have been named with tags (inlet pressure, inlet temp, outlet temp, demand, flow rate, total) selection is primitive, by pen #.
 - The property ‘tag name’ isn’t available
- The easier it is to pick and choose correctly (because items are identified as people know them), the more the ‘database’ will be exploited

How will OPC UA manifest itself?

- Apps for mobiles will extend functionality
 - Definition of OPC is “data transfer to HMI’s”
 - HMI’s are portables:
 - Tablets
 - Cell phones
 - Widgets for your desktop computer
- Expanded use through its evolution
 - The movement away from a Microsoft requirement (no more DCOM)
 - Use of XML for a data language
 - mobile O/S can exploit OPC UA, Linux O/S (XYR6000)
- Reduce the cost of developing control room HMI’s
 - Automated build by auto-population of the tag database
- OPC UA is the backbone of IIoT
 - Industrial Internet of Things



Wormhole between industrial data
and our mobile apps



Why isn't it here now?

- Reason #1 software
- Reason #2 software
- Reason #3 software
- Latest version of Wonderware is OPC UA
- What version does Abbott run? (answer: 5 year old version, not OPC UA)
- We're waiting the mobile and desktop apps which exploit OPC UA.
- Hold your breath, it's coming.



Overview of OPC DA and OPC UA

Questions?



