LISTEN.
THINK.
SOLVE.®

PharmaSuite®

PHASES OF THE EQUIPMENT AUTOMATION PACKAGE
RELEASE 8.1
TECHNICAL MANUAL

Allen-Bradley · Rockwell Software

Rockwell Automation

• Copyright

Contact Rockwell See contact information provided in your maintenance contract.

Copyright Notice © 2016 Rockwell Automation Technologies, Inc. All rights reserved.
This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

Trademark Notices FactoryTalk, PharmaSuite, Rockwell Automation, Rockwell Software, and the Rockwell Software logo are registered trademarks of Rockwell Automation, Inc.

The following logos and products are trademarks of Rockwell Automation, Inc.:

FactoryTalk Shop Operations Server, FactoryTalk ProductionCentre, FactoryTalk Administration Console, FactoryTalk Automation Platform, and FactoryTalk Security.
Operational Data Store, ODS, Plant Operations, Process Designer, Shop Operations, Rockwell Software CPGSuite, and Rockwell Software AutoSuite.

Other Trademarks ActiveX, Microsoft, Microsoft Access, SQL Server, Visual Basic, Visual C++, Visual SourceSafe, Windows, Windows Vista, Windows 7 Professional, Windows Server 2008, and Windows Server 2012 are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

ControlNet is a registered trademark of ControlNet International.

DeviceNet is a trademark of the Open DeviceNet Vendor Association, Inc. (ODVA).

Ethernet is a registered trademark of Digital Equipment Corporation, Intel, and Xerox Corporation.

OLE for Process Control (OPC) is a registered trademark of the OPC Foundation.

Oracle, SQL*Net, and SQL*Plus are registered trademarks of Oracle Corporation.

All other trademarks are the property of their respective holders and are hereby acknowledged.

Warranty This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

Contents

Introduction

This documentation contains important information how to administer, configure, and adapt phase building blocks of the Equipment Automation Phase Package including information about actions that need to be performed in FactoryTalk® ProductionCentre and PharmaSuite.

The manual is an addition to the "Technical Manual Administration" [A1] (page 17) and the "Technical Manual Configuration and Extension" [A2]-[A6] (page 17).

> **TIP**
>
> In order to use phase building blocks of the Equipment Automation Phase Package, a Live Data and, especially for the **Show historical data chart** phase, a Historian infrastructure has to be set up. For details, see "Technical Manual Installation - Enterprise Edition" [A8] (page 17).

The information is structured in the following sections:

ADMINISTRATION

In this part you will find tasks for system administrators of PharmaSuite at a customer's site.

- **Get OPC values** and **Set OPC values** phases
  - Dealing with Automation Layer Floating Point values (page 5)
- **Show historical data chart** phase
  - Configuring query templates for historical data access (page 7)
  - Configuring plot styles for historical data charts (page 10)

CONFIGURATION AND EXTENSION

In this part you will find information for system engineers who implement customer-specific configurations and extensions.

- Configuration keys
  - **Show historical data chart** phase (page 13)

Finally, the Reference Documents (page 17) section provides a list of all the documentation that is referenced in this manual.

Intended Audience

This manual is intended for system engineers, phase developers, and system administrators who maintain phase building blocks of the Equipment Automation Phase Package.

They need to have a thorough working knowledge of FactoryTalk ProductionCentre, PharmaSuite, and PharmaSuite building blocks. Thus we highly recommend to participate in a FactoryTalk ProductionCentre and PharmaSuite training before starting on the tasks described in this manual.
For more information on S88, EBR, and phase building blocks, see also "Technical Manual Developing System Building Blocks" [A7] (page 17).

Naming Recommendations

When naming your artifacts (e.g. objects in Process Designer, classes, interfaces, methods, functions, building blocks) consider the following recommendation:

- Define and make use of a vendor code consisting of up to three uppercase letters as prefix (e.g. MYC for My Company).

- Prefix your artifacts with your vendor code, separated by _.
  X_ and **RS_** are reserved for PharmaSuite.

- Observe the naming conventions defined in "Technical Manual Developing System Building Blocks" [A7] (page 17).

Typographical Conventions

This documentation uses typographical conventions to enhance the readability of the information it presents. The following kinds of formatting indicate specific information:

Bold typeface   Designates user interface texts, such as

- window and dialog titles

- menu functions

- panel, tab, and button names

- box labels

- object properties and their values (e.g. status).

Italic typeface   Designates technical background information, such as

- path, folder, and file names

- methods

- classes.

CAPITALS	Designate keyboard-related information, such as
	■ key names
	■ keyboard shortcuts.
`Monospaced typeface`	Designates code examples.

Administration

This chapter applies to the **Get OPC values**, **Set OPC values**, and **Show historical data chart** phases.

Dealing with Automation Layer Floating Point Values

This section applies to the **Get OPC values** and **Set OPC values** phases. It provides important background information on the consequences of using single (4-Byte) and double (8-Byte) precision real data types as OPC data types. See also "IEEE floating point" [D3] (page 17).

PharmaSuite refers to these real data types as **Float** and **Double**. Other common names for them are:

Single precision (4-Byte) real	**32-bit IEEE 754** floating point [D4] (page 17)	VT_R4	Float
Double precision (8-Byte) real	**64-bit IEEE 754** floating point [D5] (page 17)	VT_R8	Double

Both data types guarantee a limited precision only. Actually, it is not possible to store an arbitrary floating point value without risking to lose the ability to reconstruct the original value. **Float** guarantees a precision of **~ 7 decimal digits** and **Double** of **~ 16 decimal digits**. The bigger the number, the smaller the available fractional part (scale).

As of today, Automation Systems usually use the **Float** data type. **Double** is rarely used and not always available. On the other hand, most of the typical sensor values do not exceed the provided precision.

The following examples illustrate the behavior:
Providing **89.2** as a set point based on the **Float** data type results in **89.1999969482421875** as internal 4-Byte real representation. This is the closest value to **89.2** a 4-Byte real can handle. The internal representation does not matter. With the given number of decimal digits, the original value can be reconstructed.
However, if **89.1999969** is stored, the same internal representation is used. The reconstruction of the original value delivers **89.2**. But the number of decimal digits is clearly above the supported precision.
Contrary to the given example, **89.5** has an exact representation as a 4-Byte real.

Implications and Rounding Strategies

Ideally, **Float** data types should be avoided if higher floating point precision is desired. It is understood, that as of today most Automation Systems still use **Float** data types.

To provide a clear and consistent system behavior, PharmaSuite provides a maximum supported precision for **Float** of **7 decimal digits** and **Double** of **16 decimal digits**.

Any floating point number retrieved from an Automation System gets a precision based on the rounding to the maximum supported precision. Potential trailing zeroes are removed from the result.

Whenever PharmaSuite provides a floating point number to an Automation System the precision supported by the Automation System is not exceeded. So, a number with a precision that the target real data type cannot handle is not accepted by the respective phase building block.

PharmaSuite persists floating point tag values based on the **MeasuredValue** data type. **MeasuredValue** is an internal encapsulation based on the Java **BigDecimal** data type. The **BigDecimal** data type does not have to deal with the described limitations. It is possible to store any number with the desired precision in the supported range.

PharmaSuite defines the supported range for the **MeasuredValue** data type between **-99,999,999,999,999.999999999** and **999,999,999,999,999.999999999**. A value exceeding this range is not accepted by PharmaSuite.

Thus, for **Double** numbers with a long fractional part, an adjustment of the scale happens by rounding the number to the maximum supported scale of nine.

For more examples, see the table below:

Data type	OPC	PharmaSuite	Comment
Float	21.1	21.1	---
Float	89.2	89.2	---
Float	89.1999969	89.2	Precision exceeded
Double	89.1999969	89.1999969	---
Float	89.19999	89.19999	---
Float	89.199	89.199	---
Float	89.5	89.5	---
Float	123,456.782	123,456.8	Precision exceeded
Double	123,456.782	123,456.782	---
Float	123,456.78	123,456.8	Precision exceeded
Double	123,456.1234567899	123,456.123456790	Supported scale exceeded
Double	12,345,612.34567899	12,345,612.34567899	---

Configuring Query Templates for Historical Data Access

This section applies to the **Show historical data chart** phase. It describes how to configure query templates in FactoryTalk ProductionCentre for the historical data access. PharmaSuite uses query templates to fetch historical data values of a specific historian point and time period from a historian server.

The query templates are stored in an XML-based definition in the **QueryTemplates** FactoryTalk ProductionCentre **List** object. The **Phase/ShowHistoricalDataChartPhase/QueryTemplates** configuration key defines the **List** object to be used. For details, please refer to chapter "Configuration Keys of PharmaSuite" in Volume 4 of the "Technical Manual Configuration and Extension" [A5] (page 17).

To adapt the list of available query templates, proceed as follows:

1. In Process Designer, expand the **Lists** node and select the **QueryTemplates** list.

2. If version control is enabled, click the **Check out** button.

3. Edit the list as required.
   Example: To add a query for interpolated data, add the following query configuration before the final closing tag (**</QueryTemplates>**).
   The query references the following parameters: Property, Timestamp1, Timestamp2, and Duration; it returns the following output columns: value and time. For details of the tags and attributes formatted in red, see section "Tags and Attributes of a Query Template" (page 8).

```xml
<QueryTemplate name="Interpolated data" description="Data retrieved at regular
    time steps within defined period." usage="chart">
<Parameters>
 <Parameter name="Property" description="pi-point" datatype="String"
    systemdefined="true"/>
 <Parameter name="Timestamp1" description="start-time" datatype="DateTime"/>
 <Parameter name="Timestamp2" description="end-time" datatype="DateTime"/>
 <Parameter name=" Duration " description="time-step " datatype="Duration" />
</Parameters>
<Outputs>
 <Parameter name="value" description="numeric tag value" datatype="Float"
    systemdefined="true"/>
 <Parameter name="time" description="time stamp" datatype="DateTime"
    systemdefined="true"/>
</Outputs>
<Query provider="OSI_PI">
 SELECT if status = 0 THEN value ELSE null "value", time FROM piarchive..piinterp WHERE
    tag=%Property% AND time BETWEEN %Timestamp1% AND %Timestamp2% AND timestep =
    % Duration %
</Query>
</QueryTemplate>
```

4. Save and check in the list.

> **TIP**
>
> Before you modify an available query configuration, make sure that the configuration is not used by a phase of a master recipe or workflow. Your change may imply a change of your master recipe or workflow or may invalidate the execution of the phase.
>
> We strongly recommend to use only one system-wide setting of the **List** object for all users and stations. After changing the **List** object, a restart of Recipe and Workflow Designer and the Production Execution Client is required.

The query template used by a plot of the choice list of the **Show historical data chart** phase is specified during recipe design with the **Query template** attribute of the **Chart plot** process parameter of the **Show historical data chart** phase.

Some of the attributes of the **Chart plot** process parameter can be mapped to corresponding template parameters. The attributes serve as placeholders for different data types.
At phase execution, the formal template parameters are substituted by the actual values of the respective process parameter attributes.
For details of the mapping, see section "Process Parameter Attributes and Template Parameters" (page 9).

Tags and Attributes of a Query Template

A query template consists of the following important tags and attributes:

- Name
  The name of the query template displayed in the choice list of the **Query template** attribute of the **Chart plot** process parameter of the **Show historical data chart** phase.

- Description
  Explanatory text to describe the purpose of the query and its logic.

- Usage
  The **Show historical data chart** phase considers only templates of the **chart** usage type.

- Parameters
  Each parameter that is referenced by the query must be defined. Attributes are: name, description, datatype (One of String, Long, Float, Decimal, DateTime, Duration), and the **systemdefined** flag.
  By default, the **systemdefined** flag is set to **false** since it is intended for internal purposes only.

- Outputs
  Each result column that is returned by the query must be defined. Attributes are: name, description, datatype (One of String, Float, DateTime), and the

systemdefined flag. The flag is always set to **true**.

The **Show historical data chart** phase considers two output columns:

1. value: a float (for normal plots), string (for digital state plots), or null (if the status is not **Good**)

2. time: a DateTime value with the timestamp of the value

■ Query

For each provider, the SQL select statement is specified by referencing the parameters defined above in the **%Parameter name%** syntax. PharmaSuite supports the following providers: **OSI_PI** and **CSV_MOCK** (for internal purposes only).

Process Parameter Attributes and Template Parameters

The table shows the mapping between attributes of the **Chart plot** process parameter and the template parameters. The default query templates of PharmaSuite use only the template parameters formatted bold.

Attribute	Template parameter	Description	System-defined
Property	**%Property%**	Replaced by the name of the Historian point of the referenced Historian property type of the identified equipment entity.	Yes
N/A	**%intervalCount%**	Integer value to specify an interval count when querying plot data. The system determines a value for the highest X-axis resolution of the chart in the report.	Yes
Timestamp 1	**%Timestamp1%**	DateTime value typically used to specify the start time of a query.	No
Timestamp 2	**%Timestamp2%**	DateTime value typically used to specify the end time of a query.	No
String 1	%String1%	String, e.g. used to specify a batch number for a query.	No
String 2	%String2%	String, e.g. used to specify a unit number for a query.	No
Duration	%Duration%	A duration value, e.g. used to specify a time step value when querying interpolated data.	No
Long	%Long%	An integer value, e.g. used to specify a status when querying raw data.	No
MeasuredValue 1	%MeasuredValue1%	A decimal value (UoM is ignored), e.g. used to specify the lower limit for the returned value.	No
MeasuredValue 2	%MeasuredValue2%	A decimal value (UoM is ignored), e.g. used to specify the upper limit for the returned value.	No

Configuring Plot Styles for Historical Data Charts

This section applies to the **Show historical data chart** phase. It describes how to configure plot styles in FactoryTalk ProductionCentre for historical data charts. The PharmaSuite **Show historical data chart** phase uses plot styles to draw lines for the plots of the historical data values of a specific historian point from a historian server.
A plot style (in the following, the term **plot renderer** is used) is a named preset of color and line attributes used to draw plots and mark the legend for the respective plot.

The plot renderers are stored in an XML-based definition in the **PlotRenderers** FactoryTalk ProductionCentre **List** object. The **Phase/ShowHistoricalDataChartPhase/PlotRenderers** configuration key defines the **List** object to be used. For details, please refer to chapter "Configuration Keys of PharmaSuite" in Volume 4 of the "Technical Manual Configuration and Extension" [A5] (page 17).

To adapt the list of available plot renderers, proceed as follows:

1. In Process Designer, expand the **Lists** node and select the **PlotRenderers** list.

2. If version control is enabled, click the **Check out** button.

3. Edit the list as required.
    Example: To add a plot renderer for a thin, dotted line in grey, add the following plot renderer configuration before the final closing tag (**</plotRenderers>**). For details of the tags and attributes formatted in red, see section "Tags and Attributes of a Plot Renderer" (page 11).

```
<plotRenderer name="Grey dotted" description="Grey dotted 0.5">
 <paint red="127" green="127" blue="127"/>
 <stroke lineWidth="0.5">
  <dashArray>1.0</dashArray>
  <dashArray>2.0</dashArray>
 </stroke>
</plotRenderer>
```

4. Save and check in the list.

> **TIP**
>
> Before you modify an available plot renderer configuration, make sure that the configuration is not used by a phase of a master recipe or workflow. Your change may imply a change of your master recipe or workflow or may invalidate the execution of the phase.
>
> We strongly recommend to use only one system-wide setting of the **List** object for all users and stations. After changing the **List** object, a restart of Recipe and Workflow Designer and the Production Execution Client is required.

Tags and Attributes of a Plot Renderer

A plot renderer consists of the following important tags and attributes:

■ Name
The name of the plot renderer displayed in the choice list of the **Plot format** attribute of the **Chart plot** process parameter of the **Show historical data chart** phase.

■ Description
Explanatory text to describe the layout of the plot renderer.

■ paint
Integer values (0..255) for the **red**, **green**, **blue**, and **alpha** attributes of the line color.
For more information, please refer to the Java documentation [D1] (page 17).

■ stroke
Float value for the **lineWidth** and 0..n float values for the **dashArray** attributes used to specify patterns of painted and unpainted segments of the plot line.
For more information, please refer to the Java documentation [D2] (page 17).

Configuration and Extension

This chapter applies to the **Show historical data chart** phase. You will learn which phase-specific configuration keys are available.

For general information on configuration keys, see chapter "Managing Configurations" in Volume 4 of the "Technical Manual Configuration and Extension" [A5] (page 17).

> **TIP**
>
> Configuration keys are modified and defined in Process Designer.

Configuration Keys Specific to the Show Historical Data Chart Phase

This section describes the configuration keys specific to the phase.

Phase/ShowHistoricalDataChartPhase/HistoricalDataChartReportResolution

- **Type**: String
- **Value**: 300
- **Description**: Defines the chart resolution in DPI for batch report printout.
- **Range**: N/A

Phase/ShowHistoricalDataChartPhase/HistoricalDataChartsetPlotAntiAlias

- **Type**: Boolean
- **Value**: True
- **Description**: If the value is set to **true**, anti-aliasing is enabled for plots.
- **Range**: N/A

Phase/ShowHistoricalDataChartPhase/HistoricalDataChartsetTextAntiAlias

- **Type**: Boolean
- **Value**: False
- **Description**: If the value is set to **true**, anti-aliasing is enabled for text.
- **Range**: N/A

Phase/ShowHistoricalDataChartPhase/PlotRenderers

■ **Type**: Object - List

■ **Value**:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<plotRenderers xmlns="http://www.rockwell.com/mes/commons/base/graph">
<plotRenderer name="Black" description="Black solid 1.5pt">
  <paint red="0" green="0" blue="0"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Red" description="Red solid 1.5pt">
  <paint red="255" green="0" blue="0"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Green" description="Greed solid 1.5pt">
  <paint red="32" green="172" blue="32"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Blue" description="Blue solid 1.5pt">
  <paint red="0" green="0" blue="255"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Magenta" description="Magenta solid 1.5pt">
  <paint red="255" green="0" blue="255"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Cyan" description="Cyan solid 1.5pt">
  <paint red="0" green="223" blue="218"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Orange" description="Orange solid 1.5pt">
  <paint red="225" green="170" blue="40"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Light green" description="Light green solid 1.5pt">
  <paint red="0" green="226" blue="0"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Light blue" description="Light blue solid 1.5pt">
  <paint red="33" green="160" blue="223"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Purple" description="Purple solid 1.5pt">
  <paint red="161" green="67" blue="255"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Light red" description="Light red solid 1.5pt">
  <paint red="255" green="125" blue="125"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Petrol" description="Petrol solid 1.5pt">
  <paint red="18" green="157" blue="140"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Brown" description="Brown solid 1.5pt">
  <paint red="124" green="90" blue="76"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Tan" description="Tan solid 1.5pt">
  <paint red="179" green="175" blue="13"/>
```

```
   <stroke/>
</plotRenderer>
<plotRenderer name="Livid" description="Livid solid 1.5pt">
  <paint red="113" green="135" blue="199"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Dark red" description="Dark red solid 1.5pt">
  <paint red="192" green="80" blue="77"/>
  <stroke/>
</plotRenderer>
<plotRenderer name="Tan dash dot" description="Tan dash-dotted 1.5pt">
  <paint red="179" green="175" blue="13"/>
  <stroke lineJoin="1" miterLimit="3.0">
    <dashArray>6.0</dashArray>
    <dashArray>3.0</dashArray>
    <dashArray>1.0</dashArray>
    <dashArray>3.0</dashArray>
  </stroke>
</plotRenderer>
<plotRenderer name="Black dashed" description="Black dashed 1.5">
  <paint red="0" green="0" blue="0"/>
  <stroke>
    <dashArray>6.0</dashArray>
    <dashArray>3.0</dashArray>
  </stroke>
</plotRenderer>
</plotRenderers>
```

- ■ **Description**: Specifies the list of plot renderers available for the plot-related process parameters of the **Show historical data chart** phase.

- ■ **Range**: N/A

Phase/ShowHistoricalDataChartPhase/ProviderShortcuts

- ■ **Type**: String

- ■ **Value**: (PiConnector=OSI_PI, PiMockConnector=CSV_MOCK)

- ■ **Description**: Defines shortcuts for providers. The shortcuts can be used within QueryTemplateXML, e.g. <Query provider="OSI_PI">. Each entry must be formatted as follows: key=value, e.g. PiConnector=OSI_PI

- ■ **Range**: N/A

Phase/ShowHistoricalDataChartPhase/QueryTemplates

- ■ **Type**: Object - List

- ■ **Value**:

```
<QueryTemplates xmlns="http://www.rockwell.com/mes/commons/base/query">
  <QueryTemplate name="Raw archive data" description="All raw archive values within defined
        period." usage="chart">
    <Parameters>
      <Parameter name="Property" description="pi-point" datatype="String"
        systemdefined="true"/>
      <Parameter name="Timestamp1" description="start-time" datatype="DateTime"/>
```

```
          <Parameter name="Timestamp2" description="end-time" datatype="DateTime"/>
      </Parameters>
      <Outputs>
        <Parameter name="value" description="numeric tag value" datatype="Float"
            systemdefined="true"/>
        <Parameter name="time" description="time stamp" datatype="DateTime"
            systemdefined="true"/>
      </Outputs>
      <Query provider="OSI_PI">SELECT if status = 0 THEN value ELSE null "value", time FROM
            piarchive..picomp WHERE tag=%Property% AND time BETWEEN %Timestamp1% AND
            %Timestamp2%</Query>
  </QueryTemplate>
  <QueryTemplate name="Plot data" description="Data dedicated to plotting (trending)
            applications within defined period." usage="chart">
      <Parameters>
        <Parameter name="Property" description="pi-point" datatype="String"
            systemdefined="true"/>
        <Parameter name="Timestamp1" description="start-time" datatype="DateTime"/>
        <Parameter name="Timestamp2" description="end-time" datatype="DateTime"/>
        <Parameter name="intervalCount" description="#ofPixels" datatype="Long"
            systemdefined="true"/>
      </Parameters>
      <Outputs>
        <Parameter name="value" description="numeric tag value" datatype="Float"
            systemdefined="true"/>
        <Parameter name="time" description="time stamp" datatype="DateTime"
            systemdefined="true"/>
      </Outputs>
      <Query provider="OSI_PI">SELECT if status = 0 THEN value ELSE null "value", time FROM
            piarchive..piplot WHERE tag=%Property% AND time BETWEEN %Timestamp1% AND
            %Timestamp2% AND intervalCount = %intervalCount%</Query>
  </QueryTemplate>
  <QueryTemplate name="Marker (DigitalState)" description="All digital state string values
            for chart within defined period used as segment markers." usage="chart">
      <Parameters>
        <Parameter name="Property" description="pi-point" datatype="String"
            systemdefined="true"/>
        <Parameter name="Timestamp1" description="start-time" datatype="DateTime"/>
        <Parameter name="Timestamp2" description="end-time" datatype="DateTime"/>
      </Parameters>
      <Outputs>
        <Parameter name="value" description="string tag value" datatype="String"
            systemdefined="true"/>
        <Parameter name="time" description="time stamp" datatype="DateTime"
            systemdefined="true"/>
      </Outputs>
      <Query provider="OSI_PI">SELECT DIGSTRING(status) "value", time FROM piarchive..picomp
            WHERE tag=%Property% AND time BETWEEN %Timestamp1% AND %Timestamp2%</Query>
  </QueryTemplate>
</QueryTemplates>
```

■ **Description**: Specifies the list of query templates available for the plot-related process parameters of the **Show historical data chart** phase.

■ **Range**: N/A

Reference Documents

The following documents are available from the Rockwell Automation Download Site.

No.	Document Title	Part Number
A1	PharmaSuite Technical Manual Administration	PSAD-RM008B-EN-E
A2	PharmaSuite Technical Manual Configuration & Extension - Volume 1	PSCEV1-GR008B-EN-E
A3	PharmaSuite Technical Manual Configuration & Extension - Volume 2	PSCEV2-GR008B-EN-E
A4	PharmaSuite Technical Manual Configuration & Extension - Volume 3	PSCEV3-GR008B-EN-E
A5	PharmaSuite Technical Manual Configuration & Extension - Volume 4	PSCEV4-GR008B-EN-E
A6	PharmaSuite Technical Manual Configuration & Extension - Volume 5	PSCEV5-GR005B-EN-E
A7	PharmaSuite Technical Manual Developing System Building Blocks	PSBB-PM007B-EN-E
A8	PharmaSuite Technical Manual Installation - Enterprise Edition	PSEN-IN008B-EN-E

> **TIP**
>
> To access the Rockwell Automation Download Site, you need to acquire a user account from Rockwell Automation Sales or Support.

The following third-party documentation is available online as reference:

No.	Document Title / Web Site
D1	Java Class java.awt.Color (http://docs.oracle.com/javase/7/docs/api/java/awt/Color.html)
D2	Java Class java.awt.BasicStroke (http://docs.oracle.com/javase/7/docs/api/java/awt/BasicStroke.html)
D3	IEEE floating point (http://en.wikipedia.org/wiki/IEEE_floating_point)
D4	Single-precision floating-point format (http://en.wikipedia.org/wiki/Single-precision_floating-point_format)
D5	Double-precision floating-point format (http://en.wikipedia.org/wiki/Double-precision_floating-point_format)

Revision History

The following table describes the history of this document.

Changes related to the document:

Object	Description	Document
---	---	---

Changes related to "Introduction" (page 1):

Object	Description	Document
Introduction	**Get numeric property** phase, **Get properties** phase, and **Set numeric properties** phase are replaced by **Get OPC values** phase and **Set OPC values** phase.	1.0

Changes related to "Administration" (page 5):

Object	Description	Document
Dealing with Automation Layer Floating Point Values (page 5)	**Get numeric property** phase, **Get properties** phase, and **Set numeric properties** phase are replaced by **Get OPC values** phase and **Set OPC values** phase.	1.0

Changes related to "Configuration and Extension" (page 13):

Object	Description	Document
---	---	---