

## Tech Note 765

# Implementing Block Reads Using the OPCClient Object

All Tech Notes, Tech Alerts and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002544  
Created: March 2011

## Introduction

The OPCClient object is a DeviceIntegration (DI) object that allows access to a running OPC Data Access (DA) Server. An OPCClient object supports the following operations on I/O points for the OPC DA Server:

- Subscriptions, which are implemented via scan groups.
- Read transactions, which are implemented via block reads.
- Write transactions, which are implemented via block writes.

This *Tech Note* describes implementing Block Reads.

**Note:** This *Tech Note* assumes you have a basic understanding of Wonderware Application Server and DAServers. You should also be familiar with OPC communications. For information about OPC communications, see [Tech Note 761 OPC Connections at a Glance](#).

## Application Versions

- Wonderware Application Server 3.1 and later

## Block Read Description

A block read is a set of user-defined attributes for which you want to retrieve values in a single transaction. Instead of the I/O points being on advise, as with a scan group, the attributes are updated once per transaction. A block read must be initiated from a user or script via the BlockRead.TransactionTrigger attribute.

For this *Tech Note*, we create a basic configuration that performs a block read from a local MBTCP DAServer via the OPCClient object.

## Set Up a Basic Galaxy

Follow the steps below to create a basic galaxy and objects required for this technote:

1. Open the IDE and create a Galaxy.
2. Create an instance of the \$WinPlatform object called **Platform**.
3. Create an instance of the \$AppEngine object called **Engine**.
4. Create an instance of the \$OPCClient object called **OPC**.
5. Create an instance of the \$Area object called **Area**.
6. Create an instance of the \$UserDefined object called **Tags**.
7. In **Deployment** view, arrange the object instances as follows:

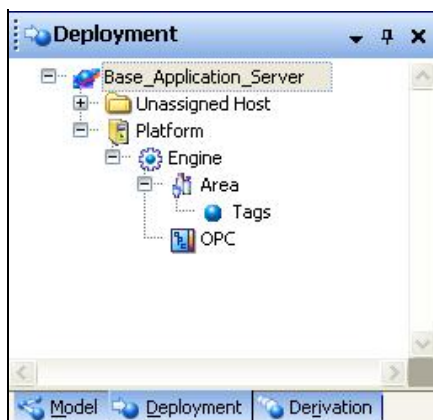


FIGURE 1: OBJECT ARRANGEMENT IN DEPLOYMENT VIEW

## Configuring DASMBTCP

**Note:** This *Tech Note* assumes you have a basic understanding of a DAServer and can configure the DAServer to communicate to a PLC.

### To configure DASMBTCP

1. Open the SMC.
2. Navigate to DASMBTCP in the DAServer manager section.
3. Create a port object called **Port**.
4. Create a PLC object called **PLC**.
5. Configure the PLC object to communicate to a PLC.

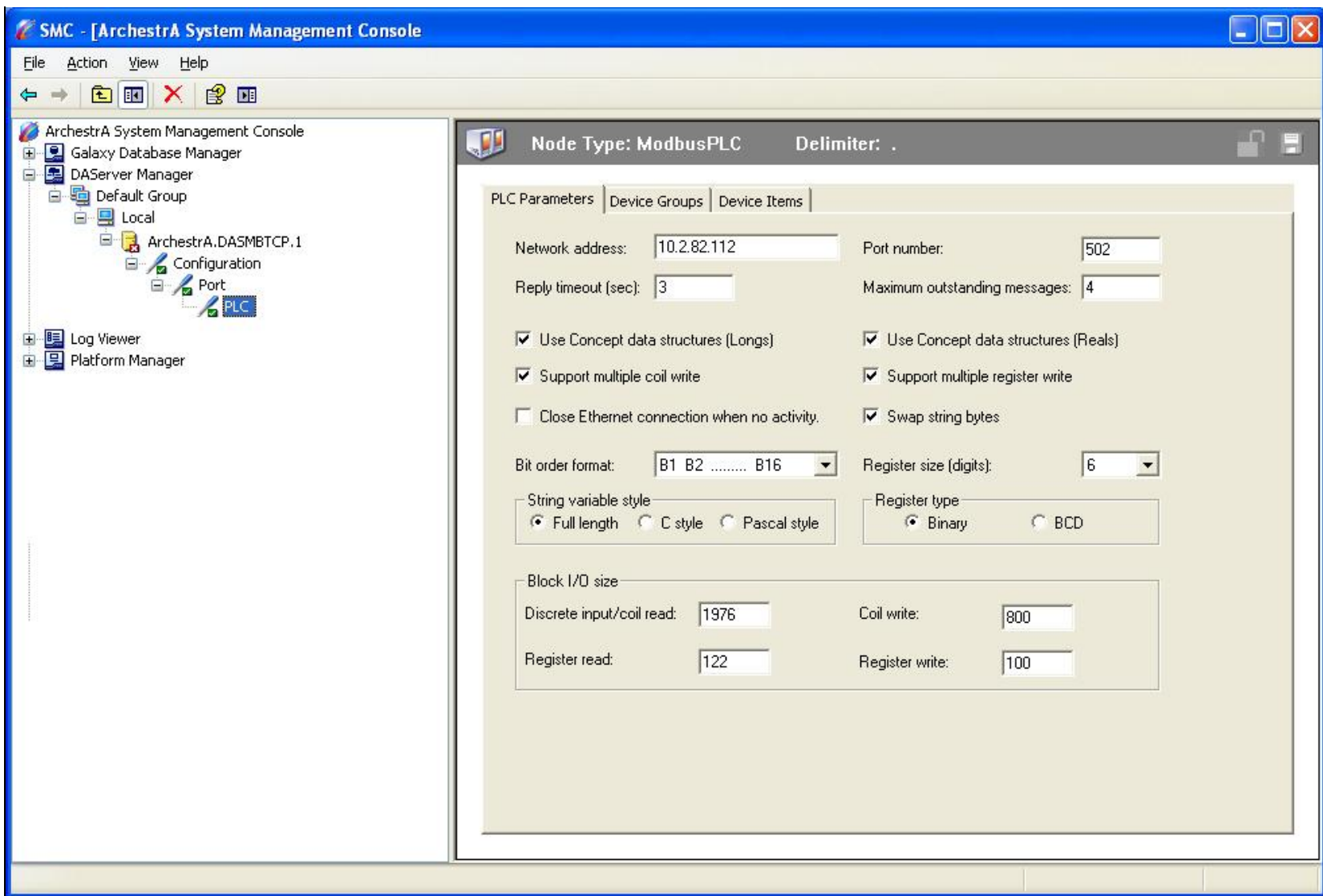


FIGURE 2: PLC OBJECT COMMUNICATION SETTINGS

## Configure the OPCClient object

1. Open the OPCClient object called **OPC**.
2. Under the **General** tab, select the **Server name**.
3. For this example, the Server name is **ArchestrA.DASMBTCP.2**. Leave all other default attribute settings.

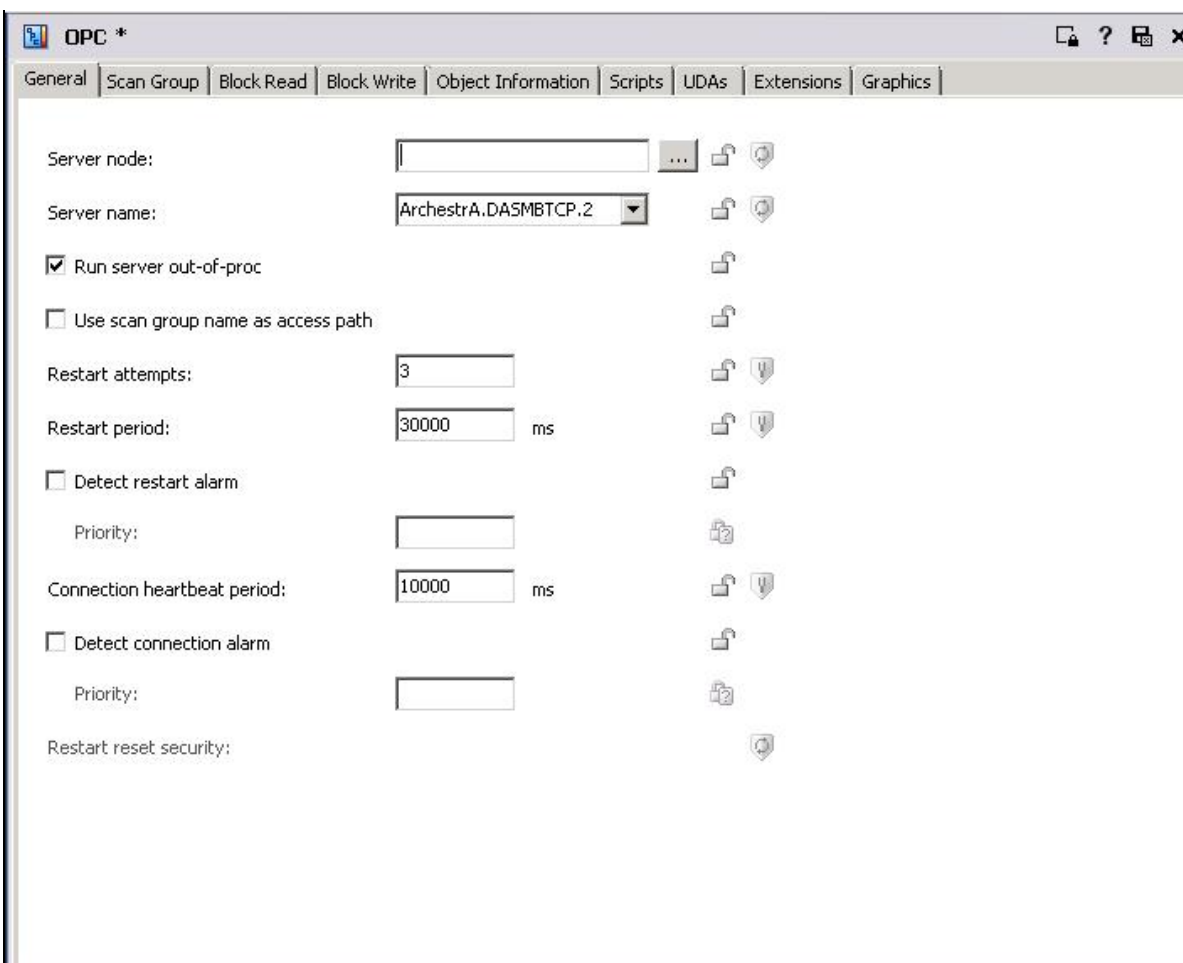


FIGURE 3: GENERAL TAB SERVER NAME AND DEFAULTS

4. Click the **Block Read** tab.

This is where we enter the block of PLC addresses that we want to read in a single transaction.

For this example, we will read addresses **400001**, **400002**, **400003**, **400006**, **400008**, and **400200**. We will also assign an attribute to each register.

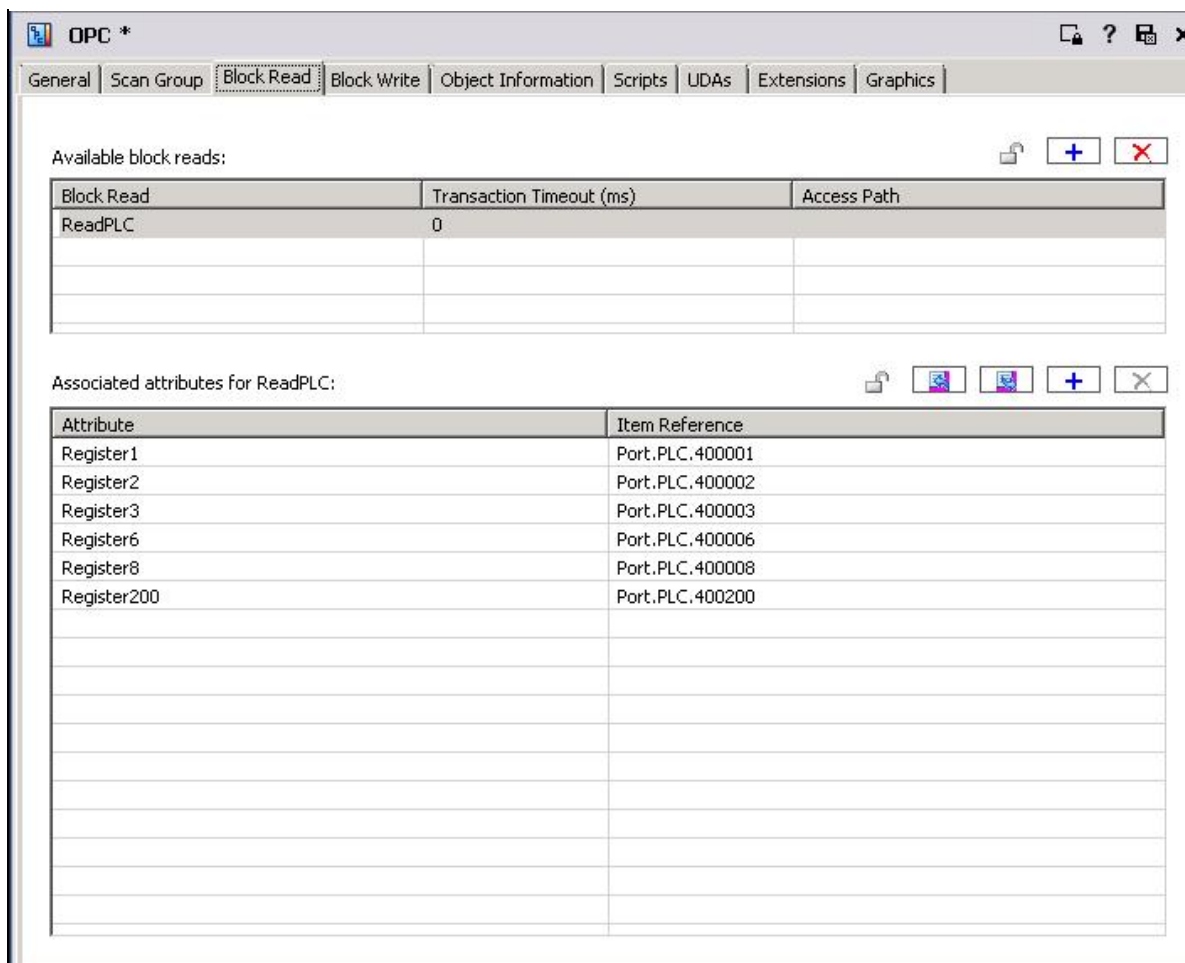


FIGURE 4: BLOCK READ ADDRESSES

**Note:** Although a Scan Group is not required for Block Reads, go to the "Scan Group" tab and enter a scan group. This will allow the saving of the OPC object without any warnings.

### Configure the UserDefined Object

1. Open the **Tags** UserDefined Object.
2. Create six analog Field attributes (Figure 5 below).
3. Provide each attribute with an input source that points to the six block read attributes as shown in the following table.

Field Attribute Name	Input Source
Tag1	OPC.ReadPLC.Register1
Tag2	OPC.ReadPLC.Register2
Tag3	OPC.ReadPLC.Register3
Tag6	OPC.ReadPLC.Register6
Tag8	OPC.ReadPLC.Register8
Tag200	OPC.ReadPLC.Register200

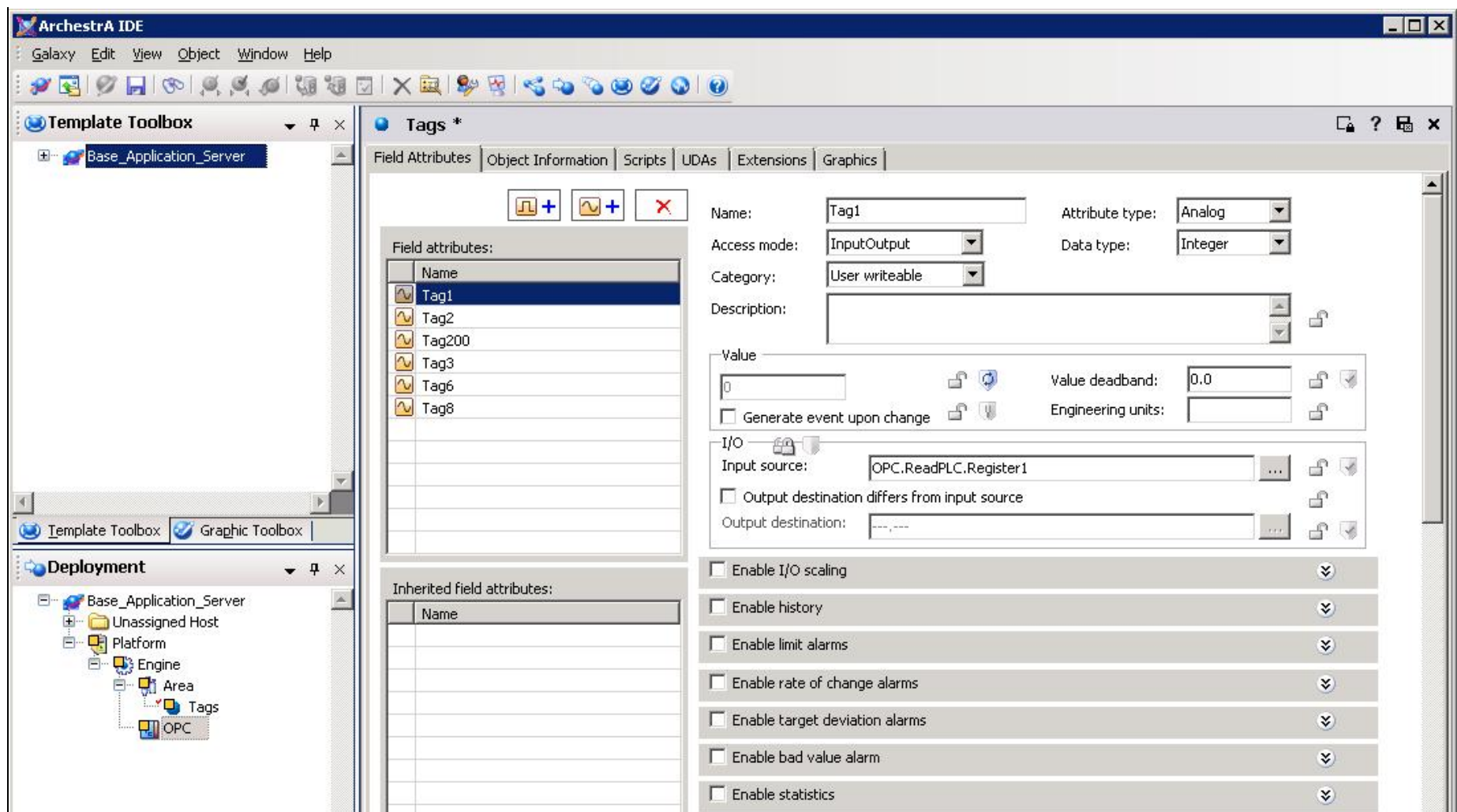


FIGURE 5: ANALOG FIELD ATTRIBUTE AND INPUT SOURCE

## Test the Configuration

1. Deploy the Platform, Engine, Area, Tags and OPC objects. After all objects are deployed, open Object Viewer.
2. Add the following attributes to the watch list:
  - Tags.Tag1
  - Tags.Tag2
  - Tags.Tag3
  - Tags.Tag6
  - Tags.Tag8
  - Tags.Tag200

- OPC.ConnectionStatus (Displays the connection status between the OPC object and the OPC Server)
- OPC.ReadPLC.TransactionTrigger (Write a "True" to trigger a block read)
- OPC.ReadPLC.TransactionStart (Displays the time and date when the block read starts)
- OPC.ReadPLC.TransactionEnd (Displays the time and date when the block read ends)

The Watch List window should look like Figure 6 (below).

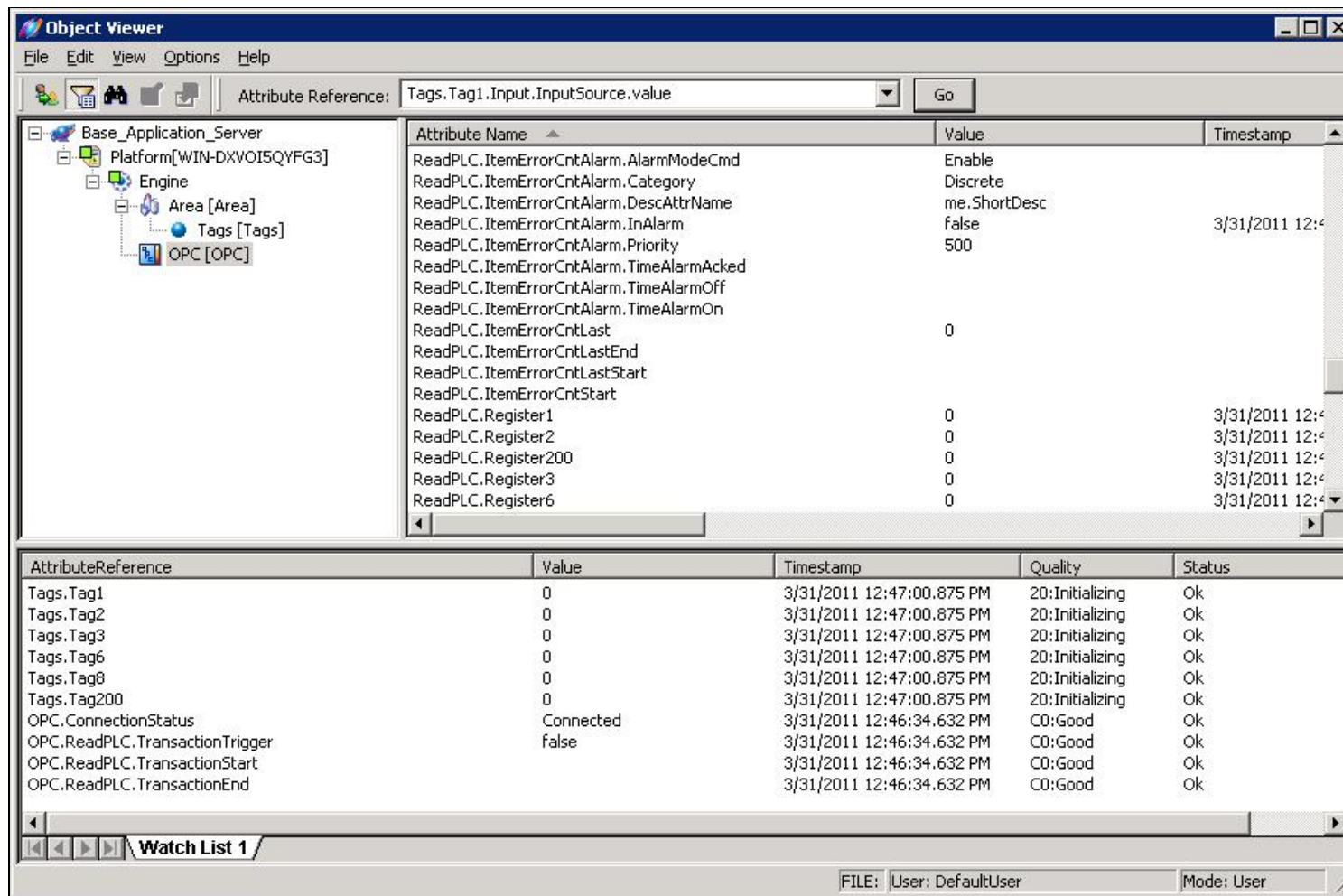


FIGURE 6: WATCH LIST WINDOW WITH OPC ATTRIBUTES

3. Trigger a block read by writing a **True** value to **OPC.ReadPLC.TransactionTrigger**.





FIGURE 7: TRIGGER BLOCK READ

If everything is configured correctly, the values are read from the PLC every time the OPC.ReadPLC.TransactionTrigger is set to **True**. OPC.ReadPLC.TransactionStart and OPC.ReadPLC.TransactionEnd will display the transaction start and end date and times (Figure 8 below).

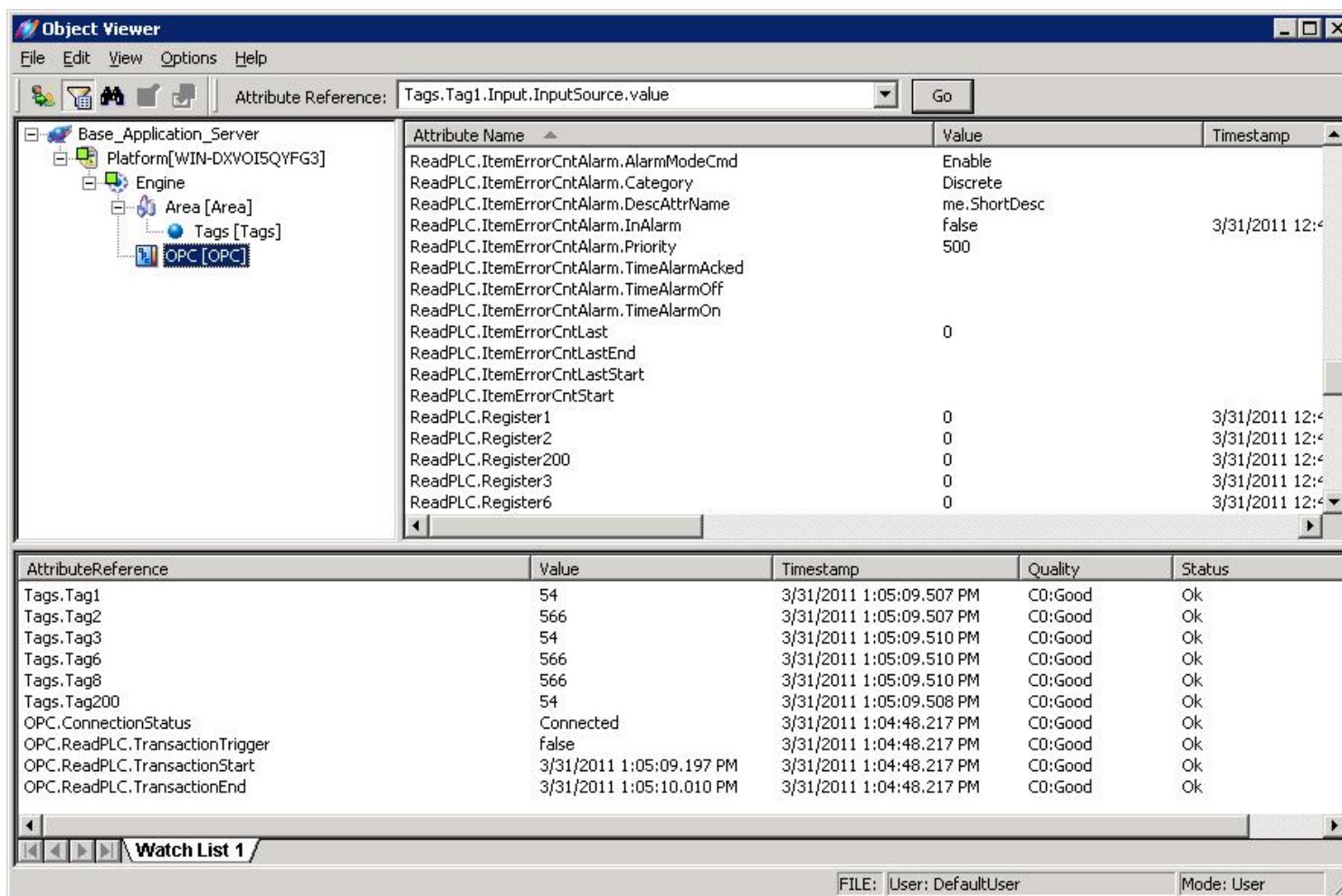


FIGURE 8: BLOCK READ WITH TIMESTAMPS

G. Allredge



software products at [Wonderware Technical Support](#).

For technical support questions, send an e-mail to [support@wonderware.com](mailto:support@wonderware.com).

 [Back to top](#)

©2011 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc. [Terms of Use](#).