WorkstationST* OPC AE Server Instruction Guide

These instructions do not purport to cover all details or variations in equipment, nor to provide for every possible contingency to be met during installation, operation, and maintenance. The information is supplied for informational purposes only, and GE makes no warranty as to the accuracy of the information included herein. Changes, modifications, and/or improvements to equipment and specifications are made periodically and these changes may or may not be reflected herein. It is understood that GE may make changes, modifications, or improvements to the equipment referenced herein or to the document itself at any time. This document is intended for trained personnel familiar with the GE products referenced herein.

Public Information – This document contains non-sensitive information approved for public disclosure.

GE may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not provide any license whatsoever to any of these patents.

GE provides the following document and the information included therein as is and without warranty of any kind, expressed or implied, including but not limited to any implied statutory warranty of merchantability or fitness for particular purpose.

For further assistance or technical information, contact the nearest GE Sales or Service Office, or an authorized GE Sales Representative.

Revised: Oct 2015
Issued: Apr 2006


* Indicates a trademark of General Electric Company and/or its subsidiaries. All other trademarks are the property of their respective owners.

We would appreciate your feedback about our documentation. Please send comments or suggestions to controls.doc@ge.com
Document Updates

<table>
<thead>
<tr>
<th>Location</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The section, Configuring DCOM</td>
<td>Updated entire section</td>
</tr>
</tbody>
</table>

Acronyms and Abbreviations

- **CoE**: Communication Center of Excellence
- **AE**: Alarm and Events
- **DCOM**: Distributed Component Object Model
- **OPC**: A standard for data exchange in the industrial environment
- **SoE**: Sequence of Events
# Contents

1 Introduction ................................................................................................................................. 4
2 Alarm and Event Routing ............................................................................................................... 4
3 OPC AE Server Variable Name Space .......................................................................................... 5
   3.1 Use Alias Names Property ...................................................................................................... 5
4 Variable State Translations ......................................................................................................... 7
5 Language Support ......................................................................................................................... 8
   5.1 Use Second Language Property ............................................................................................. 8
   5.2 Client Connections ................................................................................................................... 9
6 Attribute Support .......................................................................................................................... 10
7 Data Update Rates ......................................................................................................................... 10
8 Configuring the OPC AE Server .................................................................................................. 11
9 Configuring DCOM ......................................................................................................................... 14
   9.1 DCOM Default Properties ...................................................................................................... 15
   9.2 Default Access Permissions .................................................................................................... 17
   9.3 Windows Workgroups Example ............................................................................................. 20
   9.4 Change OPC AE Server DCOM Settings ................................................................................. 29
   9.5 DCOM Security ....................................................................................................................... 30
10 Terms ........................................................................................................................................... 33
1 Introduction

OPC® is an industry standard for communication between vendors in an industrial environment. The non-profit OPC Foundation provides the specification for the standard, as well as programming proxy stubs for OPC. OPC Alarms and Events (AE) is the standard for real-time alarm and event data. The OPC server is OPC AE 1.1 compliant and is verified using the OPC Foundation Compliance Test Tool. OPC AE server, configured in a WorkstationST component requires the ToolboxST* application.

Note ControlST® software suite version 4.0 or newer no longer supports Windows® 2000.

2 Alarm and Event Routing

The following figure displays alarm and event routing. The OPC AE server is always connected to the local Alarm Server. The Alarm Server is required to be running.
3 **OPC AE Server Variable Name Space**

When an alarm or event is received by the Alarm Server, it is sent to the OPC AE server through an internal client connection. Before connected OPC AE clients are notified, variable name and condition states must be translated.

*Note* Variable is used to describe an entity within the ToolboxST application. *Source* is the comparable OPC AE term; the comparable OPC DA term is *Item*.

The OPC AE server alarm/event name space is mapped from the variable name space in the control system. This name space can take two forms: standard variable names or alias names. The WorkstationST component configuration provides the *Use Alias Names* property to enable the use of alias names instead of variable names. The OPC AE server provides alias names if this property is true and variable names if this property is false. The default value of this property is false.

### 3.1 Use Alias Names Property

With the *Use Alias Names* property set to false, the defined variable name space is used as follows:

\[ \text{<Alarm Type>.<Device Name>.<Variable Name>} \]

*Alarm Type* is one of the five defined types in the control system: Alarm, Event, Hold, SOE, and Diagnostic.

*Device Name* is the name of the device where the alarm or event originated.

*Variable Name* is the name of the variable as used in the ToolboxST configuration.

Example:

Device Name = G1

Variable Name = *CoolantOverTemperature* and is assigned a process alarm

The variable name used is *Alarm.G1.CoolantOverTemperature*.

*Note* The format of the name space has changed starting with the ControlST software suite version 4.0. This change is necessary due to the implementation of Alias names. *Alarm Type* is required because a single variable in the ToolboxST application can be configured as an *Alarm, Event*, and a *Hold* simultaneously. The *Alarm Type* is used to create a uniquely defined name.

*Note* The browser implementation returns alarms and events using this name space.
With the *Use Alias Names* property set to true, the alias name space is used as shown:

\[
\text{Alarm Type}.[\text{Device Name}].\text{Alias Name}
\]

where:

**Alarm Type** is one of the five defined types in the control system: Alarm, Event, Hold, SOE, and Diagnostic.

**Device Name** is the name of the device where the alarm or event originated. The use of the name is optional and is controlled by the *Enable Alias Prefix* property for each device configured in the ToolboxST application. If this property is set to true, the device name is used. If set to false, no device name is used.

**Alias Name** is the name entered in the Alias column in the ToolboxST application for each variable defined.

---

**Note** If the *Alias Name* is blank, the fully qualified variable name is used instead.

---

**Note** The browser implementation returns alarms and events using this name space.

Example:

Using the previous example, the alias name assigned to the variable is

10LBA70CT001YA04.

*Alarm. 10LBA70CT001YA04* (If *Enable Alias Prefix* is set to false.)

*Alarm.G1. 10LBA70CT001YA04* (If *Enable Alias Prefix* is set to true.)
# Variable State Translations

The OPC AE server has three classifications for alarm and events:

- **Simple** is an event that has no state defined.
- **Condition** is an event that has one or more states.
- **Tracking** is an event used for operator set point change notification and has no state defined. There is no mapping in the control system and it is not used.

The control system alarms and events are mapped as displayed in the following table.

## Alarms and Events Mapping

<table>
<thead>
<tr>
<th>OPC AE Server Classification</th>
<th>OPC AE Category Name</th>
<th>OPC AE Condition. SubCondition Name</th>
<th>Control System Alarm Type</th>
<th>Control System State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>Event</td>
<td>—</td>
<td>Event</td>
<td>—</td>
</tr>
<tr>
<td>Simple</td>
<td>SOE</td>
<td>—</td>
<td>SOE</td>
<td>—</td>
</tr>
<tr>
<td>Tracking</td>
<td>Tracking</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Condition</td>
<td>Alarm</td>
<td>Alarm.Active</td>
<td>Alarm</td>
<td>Active</td>
</tr>
<tr>
<td>Condition</td>
<td>Alarm</td>
<td>Alarm.Normal</td>
<td>Alarm</td>
<td>Normal</td>
</tr>
<tr>
<td>Condition</td>
<td>Hold</td>
<td>Hold.Active</td>
<td>Hold</td>
<td>Active</td>
</tr>
<tr>
<td>Condition</td>
<td>Hold</td>
<td>Hold.Normal</td>
<td>Hold</td>
<td>Normal</td>
</tr>
<tr>
<td>Condition</td>
<td>Hold</td>
<td>Hold.Override</td>
<td>Hold</td>
<td>Overriden</td>
</tr>
<tr>
<td>Condition</td>
<td>Diagnostic</td>
<td>Diagnostic.Active</td>
<td>Diagnostic</td>
<td>Active</td>
</tr>
<tr>
<td>Condition</td>
<td>Diagnostic</td>
<td>Diagnostic.Normal</td>
<td>Diagnostic</td>
<td>Normal</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.Normal</td>
<td>Alarm</td>
<td>Normal</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.HI</td>
<td>Alarm</td>
<td>HI</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.HI HI</td>
<td>Alarm</td>
<td>HI HI</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.LOW</td>
<td>Alarm</td>
<td>LOW</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.LOW LOW</td>
<td>Alarm</td>
<td>LOW LOW</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.LOW LOW LOW</td>
<td>Alarm</td>
<td>LOW LOW LOW</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.DEVIATION HI</td>
<td>Alarm</td>
<td>Deviation</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.INHIBITED</td>
<td>Alarm</td>
<td>Inhibit</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.ROC HI</td>
<td>Alarm</td>
<td>Rate</td>
</tr>
<tr>
<td>Condition</td>
<td>Analog_Alarm</td>
<td>Analog_Alarm.BAD QLTY</td>
<td>Alarm</td>
<td>BadQlty</td>
</tr>
</tbody>
</table>

The OPC AE server implements and acknowledges Alarms, Diagnostics, and Holds as condition-related states.

**Note** Events and SOEs are simple events, and cannot be acknowledged from any external OPC AE client.
5 Language Support

The OPC AE server implements the same language set (primary and secondary) as defined in the ToolboxST application. There are two ways to handle getting notifications in these languages: the Use Second Language property and client connections.

5.1 Use Second Language Property

This property is defined in the WorkstationST component's General tab Property Editor. When set to true, the OPC AE server sends all notifications to all clients in the Second Language.

The fields that are affected by the language selection are:

Description is the second language description assigned to the variable in the ToolboxST application. If the second language description is blank then no description is returned.

Alarm States is the current alarm state. The value returned follows these rules:

- If the Use Second Language property is true then:
  - If the alarm state for the second language is NOT blank the alarm state second language text is used.
  - If the alarm state for the primary language is NOT blank the alarm state primary language text is used.
  - Otherwise the default alarm state text is used.

- If the Use Second Language property is false then:
  - If the alarm state for the primary language is NOT blank the alarm state primary language text is used.
  - Otherwise, the default alarm state text is used.
5.2 Client Connections

The client can query the server to determine what locales (languages) are supported. The client then selects one of the locales and sets the locale in the server. The OPC AE server returns all alarm and event notifications using the selected language to the client.

**Note** If the Use Second Language property is true, this causes the client to receive notifications in the second language regardless of the client's selected locale.

The fields that are affected by the client's language selection are:

**Description** is the second language description assigned to the variable in the ToolboxST application.

- If the locale selected is the primary language or is the invariant language, the description assigned to the variable in the ToolboxST application is used.
- If the locale selected is the second language, the second language description assigned to the variable in the ToolboxST application is used. If the second language description is blank then no description is returned.

**Alarm States** is the current alarm state. The value returned follows these rules:

- If the client has selected the second language then:
  - If the alarm state for the second language is NOT blank the alarm state second language text is used.
  - If the alarm state for the primary language is NOT blank the alarm state primary language text is used.
  - Otherwise the default alarm state text is used.
- If the client has selected the primary or invariant language then:
  - If the alarm state for the primary language is NOT blank the alarm state primary language text is used.
  - Otherwise the default alarm state text is used.
6 Attribute Support

When a client connects to the OPC AE server, the client can query the server to determine the number of attributes that are supported. One of the attributes is the Alias attribute. The client can select these attributes to be returned with the alarm and event notifications when the subscription is defined. The attributes available to the client are:

**Alias** is an alternate name defined for the variable.

**Primary Language Description** is the description of the variable.

**Secondary Language Description** is the alternate description for the variable.

**Service State** is the current service state of the variable. The values are:

- 0 = Undefined
- 1 = In Service
- 2 = Out of Service

**Shelved State** is the shelved state of the variable. The values are:

- 0 = Undefined
- 1 = Unshelved
- 2 = Shelved

**Shelved Time** is the time the alarm was shelved.

**Service Time** - The time the variable was placed into or out of service.

7 Data Update Rates

When a client connects to the OPC AE server, an update notification rate can be specified. The OPC AE server imposes a restriction on this rate and this restriction is returned when the subscription is created by the client. A maximum of 1024 events are sent in one notification.
8 Configuring the OPC AE Server

The ToolboxST application version 2.2 or higher is used to configure OPC AE server features. The configuration properties are defined by opening a WorkstationST component in the system.

The OPC AE server is configured as a feature of the WorkstationST component.

➢➢ To configure the OPC AE server

1. From the Windows Start menu, select Programs, GE ControlST, ToolboxST, and ToolboxST to display the System Editor.

2. From the Tree View, double-click a WorkstationST component to display the Component Editor.

3. From the General tab Tree View, highlight the Features item.

4. From the Summary View, click and set the Enabled column to True for the Alarm Server and the OPC AE Server features. The Alarms and OPC AE Server tabs display.

Note The Alarm Server must be enabled for the OPC AE server to be used.
5. From the **Tree View**, click the **Consumed Devices** item and select all components from which the Alarm Server and OPC AE server will receive alarms and events.

![Image of Tree View with Consumed Devices selected]

6. Click the **Alarms** tab to verify that the necessary options have been set. There are no options defined here that affect the operation of the OPC AE server.

7. From the **OPC AE Server** tab, select **Publish Settings**. All alarm and event types are published by default. These settings are for each consumed device.

*Note* Changing these settings will result in these alarms not being sent to external OPC AE clients in the control system. It does not affect alarms or events displayed in the WorkstationST Alarm Viewer.
The WorkstationST component can be built and downloaded. Use the status monitor to check the running health of the OPC AE server. The DCOM configuration settings may need to be modified for the GeCssOpcAeServer application before external clients can successfully connect.
9 Configuring DCOM

The Distributed Component Object Model (DCOM) utility allows components to communicate across network boundaries but is also involved with client to server interaction on the same computer. DCOM is configured for both the server and client computers using dcomcnfg.exe.

**Note** The DCOM utility resides in the Windows System32 directory.

➢ To start the DCOM utility

1. From the Windows Start menu select All Programs, Accessories, and Run.
2. In the Run dialog box, type `dcomcnfg.exe`, and click OK. The Component Services window displays:
9.1 DCOM Default Properties

DCOM must be configured to allow the client user access to the server computer, and the server user access to the client computer. The server user is the system account on the server computer. Adding DOMAIN\ComputerName into the access permissions allows access by the server to the client.

**Note** This does not apply to computers using Windows workgroups. Refer to the section, *Windows Workgroups Example*.

➢ To configure default properties: from the Component Services screen, right-click My Computer and select Properties.
From the My Computer Properties dialog box Default Properties tab, click to select Enable Distributed COM on this computer.

Click OK.

This configuration is the default. The Default Authentication Level on the client computer should either match, or be more restrictive than the authentication level on the server. When a DCOM connection is attempted, the higher of the two levels is used. If the server is configured for Connect level, and the client is configured for None, the client is rejected. This authentication process occurs before any other DCOM security is checked.
9.2 Default Access Permissions

The Default Access Permissions allow a remote client to communicate with the server. Communication between the client and the server is required for connecting, adding OPC groups, and browsing variables. The client computer must allow access by the server for live values to be updated.

Note Windows defaults the access permissions to allow access for both system and self. To allow any client to connect, you must add Interactive with Allow Access permissions to the Default Access permissions.

The server is configured to run as a service and, by default, runs as a system. To receive live data updates, the client computer must allow the system account from the server computer remote access.

➢➢ To edit the Default Access Permissions

From the My Computer Properties dialog box, click the COM Security tab.
Click **Add** to display the **Select Users, Computers, Service Accounts, or Groups** dialog box.

Click **Object Types** to display the **Object Types** dialog box.

**Note:** If the computers are in a domain, you can add **Object Types** of **Computer**. If the computers are in a workgroup, this feature is not available.
Click to select **Computers**.

Click OK.

Enter the computer name and click **Check Names** to verify the computer exists in the domain.

Enter your credentials if prompted.

Click OK.

In the above example, the computer named Corsair contains the OPC server. Corsair is added with access to this computer.

Add the same computer setting to the Limits for Access, Limits for Launch and Activation, and to Default for Launch and Activation. Repeat this procedure for both Client and Server computers.

If the logon was changed to a different user, add the user computer rather than the server computer (refer to the section, *Changing the OPC AE Server DCOM Settings*).
9.3 Windows Workgroups Example

This section describes the settings required to connect a DCOM client running as the System account to a DCOM server running as the system account on a remote computer. Services run as the system account.

**Note** The System user is not the same as the Administrator user.

When a client running as System tries to connect to another computer in a workgroup, that client has no network credentials. If the computers were in the same Windows domain, the client System user can be identified, but when using workgroups, the remote server computer cannot identify the client user. Under these conditions, the client is seen by the server as Anonymous Logon user.

**Note** Permissions must be applied to the server computer to allow the client to communicate to the server (connect, browse, read, write). For the server to respond with data change notifications, the settings must be applied to the client computer.

Ensure that the *Authenticate Users as Themselves* local security policy has been set correctly.

Both the computers must be in the same workgroup and have an identical account and password on each. This common account is the account under which the OPC DA client runs. This account should be included in the Default Access and Default Launch and Activation Privileges with Remote Access enabled.

The default properties of the computer are left as the Windows default. For information on running dcomcnfg.exe and changing computer properties, refer to the section, Configuring DCOM.
9.3.1 Set Security Limits and Defaults

➢ To set security limits

1. From the Windows Start menu select All Programs, Accessories, and Run.

2. In the Run dialog box enter dcomcnfg.exe and click OK. The Component Services window displays.

3. From the Component Services screen, right-click My Computer and select Properties.
From the **COM Security** tab, in the **Access Permissions** section, click **Edit Limits** to display the **Access Permission** dialog box.
Verify the Allow check boxes for both Local and Remote Access are selected for each user or group.

**Note** If ANONYMOUS LOGON is not on the list of Group or user names, refer to the section *Add an Anonymous User* to add it.

Repeat these steps for Edit Default in Access Permissions, Edit Limits and Edit Default in Launch and Activation Permissions, verifying that all Allow check boxes are selected for each user or group.
### 9.3.2 Add an Anonymous User

The OPCEnum service provided by the OPC Foundation does not include the ANONYMOUS LOGON, so it must be added to the Launch and Access permissions in the OPCEnum Properties dialog box. This change does not require a restart. The settings take effect the next time the OPCEnum service is started. For information on changing an individual DCOM server’s settings, refer to the section, *Changing the OPC Server DCOM Settings*.

➢➢➢

To add users and permissions

1. From the Windows **Start** menu select **All Programs, Accessories**, and **Run**.
2. In the **Run** dialog box enter `dcomcnfg.exe` and click **OK**. The **Component Services** window displays.
3. From the **Component Services** window, expand the **DCOM Config** folder.

**OpcEnum** displays after the WorkstationST application is installed.

Right-click **OpcEnum** and select **Properties** to display the **OpcEnum Properties** dialog box.

---

![Component Services Window](image-url)
From the **Security** tab, the default setting for **Launch and Activation Permissions** is **Customize**.

Click **Edit** to display the **Launch and Activation** permission dialog box.

Learn more about [setting these properties](#).
In the text box, enter **anonymous logon** and click **OK**.

The **Launch and Activation Permission** dialog box changes to display **ANONYMOUS LOGON** in the **Group or user names** text box.
Select **ANONYMOUS LOGON**.

Select the **Allow** checkboxes for **Local and Remote Launch** and **Local and Remote Activation**.

Click **OK**.
To set *Access Permissions* and *Configuration Permissions*, return to the **OpcEnum Properties** dialog box **Security** tab and repeat the previous three steps for each section.
9.4 Change OPC AE Server DCOM Settings

The default settings for the OPC AE server are normally adequate. However, if the OPC AE server needs to run as a user, select Control Panel, Administrative Tools, and Services to modify the Logon As (user). This setting is configured during installation, so installing a new version runs the service as the default System account again.

**Note** If you change the Logon As setting, the DCOM identity setting must be configured to match.

➢➢➢ To change the GeCssOpcAeServer identity:

1. From the Component Services screen Tree View, expand Console Root, Component Services, Computers, and DCOM Config.
2. Right-click the GeCssOpcServer item and select Properties to display the GeCssOpcServer Properties dialog box.
3. Click the Identity tab, verify that the option The system account (services only) is selected and click OK.
9.5 DCOM Security

The following information is an excerpt from www.opcfoundation.org:

9.5.1 Abstract

OPC server vendors have two approaches to networking:

- The client can connect to a local server to use the existing proprietary network scheme. This approach will commonly be used by vendors who are adding OPC capability to an existing distributed product.
- The client can connect to the desired server on a target machine, then use DCOM for networking. This approach may be used in conjunction with the above approach.

Using DCOM for remote OPC client/server communications is necessary for cross-vendor interoperability. Consequently, there are several issues that surface in the design, development, implementation, and deployment of distributed (DCOM-enabled) OPC components.

DCOM can make distributed applications secure without any security-specific coding or design in either the client or the component. Just as the DCOM programming model hides a component's location, it also hides the security requirements of a component. The same (existing or off-the-shelf) binary code that works in a single-machine environment, where security may be of no concern, can be used in a distributed environment in a secure fashion.

DCOM achieves this security transparency by letting developers and administrators configure the security settings for each component. Just as the Windows NT File System lets administrators set access control lists (ACLs) for files and directories, DCOM stores Access Control Lists for components. These lists simply indicate which users or groups of users have the right to access a component of a certain class. These lists can easily be configured using the DCOM configuration tool (DCOMCNFG) or programmatically using the Windows NT registry and Win32® security functions.

Whenever a client calls a method or creates an instance of a component, DCOM obtains the client's current username associated with the current process (actually the current thread of execution). Windows NT guarantees that this user credential is authentic. DCOM then passes the username to the machine or process where the component is running. DCOM on the component's machine then validates the username again using whatever authentication mechanism is configured and checks the access control list for the component (actually for the first component run in the process containing the component).

If the client's username is not included in this list (either directly or indirectly as a member of a group of users), DCOM rejects the call before the component is ever involved. This default security mechanism is completely transparent to both the client and the component and is highly optimized. It is based on the Windows NT security framework, which is probably one of the most heavily used (and optimized!) parts of the Windows NT operating system: on each and every access to a file or even to a thread-synchronization primitive like an event or semaphore, Windows NT performs an identical access check. The fact that Windows NT can still compete with and beat the performance of competing operating systems and network operating systems shows how efficient this security mechanism is.

There are three main issues: authentication, launch (activation) permission, and access (call) permissions, which all operate more or less independently of each other.
The first thing Windows NT does is to authenticate the user (as in the figure above). Whether or not this is done depends on the authentication level defined in DCOMCNFG. This level is specified by both the client and server machines: the server specifies the minimum required authentication level for incoming calls (any call that comes in below this is automatically rejected via E_ACCESSDENIED), and the client specifies its required authentication level for each interface call. COM automatically uses the higher of the two settings. More information on these settings can be found in the HELP file for DCOMCNFG.

Once the user has been authenticated, two additional types of security are defined in DCOM: activation security (permissions) and call security (permissions).

Activation security controls which classes a client is allowed to launch and retrieve objects from, and is automatically applied by the Service Control Manager of a particular machine. Upon receipt of a request from a remote client to activate an object, the Service Control Manager of the machine checks the request against activation setting information stored within its registry.

The HKEY_LOCAL_MACHINE\Software\Microsoft\OLE key's DefaultLaunchPermission named value sets the machine-wide default access control list (ACL) to specify who has access to classes on the system. For class-specific activation settings (which take precedence over the default setting), the HKEY_CLASSES_ROOT\APPID\{…} key’s LaunchPermission named value contains data describing the class’s ACL. These keys are set initially when NT is installed, and can be modified by dcomcnfg.exe.

Call security provides the security mechanism on a per-call basis that validates inter-object communication after a connection between a client and server has been established. Call security services are divided into three categories:

9.5.2 General Functions Called by Both Clients and Servers

- New interfaces on client proxies
- Server-side functions and call-context interfaces

The HKEY_LOCAL_MACHINE\Software\Microsoft\OLE key’s DefaultAccessPermission named value sets the machine-wide default access control list (ACL) to specify who has access to classes on the system. For class-specific activation settings (which take precedence over the default setting), the HKEY_CLASSES_ROOT\APPID\{…} key’s AccessPermission named value contains data describing the class’ ACL. These keys are set initially when NT is installed and can be modified by dcomcnfg.exe.
DCOM Overview

1. DCOM obtains user name

2. Authenticate User

3. Is User in the list?

4. Fail or allow call

Access Control List for Component:
BillG
MyAppUserGroup

(Extended) Directory:
BillG@Password,
SteveB@Password
MyAppUserGroup = 
(SteveB), Administrator.

Client A
User: BillG

Client B
User: SteveB

Client B
User: JohnSmith

Component

Security Provider

DCOM

DCOM
10 Terms

**Areas** - A grouping of alarm/events in a hierarchical way that represents the physical organization of the equipment generating the alarms/events.

**Feature** - An element of the WorkstationST runtime system, which can be optionally enabled through ToolboxST. Examples include OPC DA server, Recorder, OPC AE server and Alarm System.

**Notification** - The term used to describe a set of alarm/events being sent from the OPC AE server to the connected client.

**OPC** - A standard for data exchange in the industrial environment. The OPC foundation provides specifications for various OPC standards such as OPC DA (Data Access) and OPC AE (Alarm and Event).

**OPC Alarm and Events** - Used in the context of OPC AE Clients/OPC AE servers that means a collection of event notifications and their conditions.

**OPC Event** - Is a detectable occurrence which is of significance to the owning device, the OPC AE server and its clients.

**OPC Alarm** - Is the name given to an abnormal condition of an event. This event is usually associated with one or more states representing the condition.

**OPC Event Server** - Is another name for an OPC AE server.

**Runtime** - Software stored in the controller’s Flash memory that converts application code (pcode) to executable code.

**Severity** - A numeric value assigned to the alarm/event that represents the significance of the alarm/event. Defined to be in the range between 1 and 1000 where 1 is informational and 1000 is catastrophic.

**Subscription** - The term used when a client connects to the OPC AE server and specifies the update rate and filter criteria to use when the OPC AE server sends event notifications to the client. A client can have one or more subscriptions to an OPC AE server.

**TCS** - Turbine Control System. Used to differentiate the part of the system managed and configured by ToolboxST versus parts of the system that are managed by external tools and devices.