

## What is new in Version 3.12

### OPC Toolbox C++

The following points have been implemented in the OPC Toolbox C++ V3.12 since the V3.10:

#### SOCmn (Common Library)

- The format method of the class SOCmnString had problems with very large (> 1000 bytes) parameter values.
- The FADF\_VARIANT flag is now set for VT\_DECIMAL variants.
- The class SOCmnList was extended by the method removeAt, which includes the position of the entry to remove and it's predecessor.
- The class SOCmnVariant was extended by the methods setValue, getValue and getSize.
- For boolean values the class SOCmnVariant now provides the value -1 for true instead of the 1 before.
- The SOCmnObjectLock class changed, that a sequence of locks in a row to different objects will be able to create a dead lock like before.
- The SOCmnString class was extended by several methods for string handling.

#### SOSrv (Server specific Library)

- The diagnostic interface of the OPC Server Toolkits was enhanced by the functionality to set trace parameters and deliver the trace messages to a client.
- The Visual Studio 6 Wizard now generates different names for the COM server objects if OPC DA and AE are supported in one server.
- For InProc servers the diagnostic interfaces is not provided anymore, because it's not useful for this kind of server. A InProc server is implemented in a DLL and will be loaded from every client into it's process.
- The methods for the monitoring of the connections to the clients (SOSrvEntry::setCheckParameter, SOSrvServer::checkClientConnection) now include a parameter to specify if all interfaces to the client should be checked.
- The SOSrvEventHandler class now includes a method which is called every time a OPC COM object has been released.
- Applications with complete string-based namespaces can increase the performance of browse requests by implementing the method SOSrvBrowser::browseOnlyStringBased.

#### SODaS (OPC Data Access Server Library)

- At the advise of a notification interface, the toolkit now checks the activation and enable state of a group before sending the current data to the client.
- The method SODaSRequest::complete was enhanced by the parameter writeChangeCache. Via this parameter the application can control if the value of the cache is updated with the successfully written value.
- The OPC COM functions of the IOPCItemProperties interface now support multiple server instances.
- The method SODASItemTagIConnect::notifyAllConnectedObjects was enhanced by the parameter result. With this parameter the server can send I/O errors in event driven I/O to the clients.
- You can now free the existing and create a new root object of the namespace tree during the runtime of the server (SODASEntry::createNewNameSpaceRoot). This functionality is useful for the reconfiguration of a server.
- The synchronous I/O requests now use the class SODaSTransaction.
- The reaction on RPC errors when calling the notification methods of the OPC clients was improved. In this context the method disconnectClient was added.
- The toolkit now supports the external (outside of the toolkits namespace tree) management of the ItemTag and node objects of the namespace.

### **SOAeS (OPC Alarms and Events Server Library)**

- The default implementation of the method SOAeSCondition::getSeverity now provides for the inactive state the severity 1 described in the V1.10 of the AE specification. The default severity values of SOAeSConditionClassSingleStateDefault::m\_activeSeverity, SOAeSSubConditionClass::m\_severity, SOAeSSubscriptionFilterFull::m\_lowSeverity, SOAeSEventAttrSimple::m\_severity have been set to 1.
- When calling the event callback IOPCEventSink::OnEvent the actor id of condition related events now is only set if it's the confirmation of an acknowledgement (OPC\_CONDITION\_ACKED is set in wNewState).
- You can now free the existing and create a new root object of the event- and areaspace tree during the runtime of the server (SOAeSEntry::createNewEventSpaceRoot, SOAeSEntry::createNewAreaSpaceRoot). This functionality is useful for the reconfiguration of a server.

### **SOClt (Client specific Library)**

- The parameters for the automatic monitoring of the connection to the server now can set directly via the method SOCltServer::addConnectionMonitor. The connection monitoring could be stopped by calling the method SOCltServer::removeConnectionMonitor.
- The client toolkits now can be initialized (SOCltEntry::initialize) and terminated (SOCltEntry::terminate) several times in a row.
- The method SOCltCreator::createTaskConnectionMonitor makes it easier to use an own connection monitor class.
- The new example SODaC\connectionMonitor demonstrates the use of the connection monitor.

### **SODaC (OPC Data Access Client Library)**

- The performance of explicit called I/O operations in the class SODaCGroup was improved.
- Exceptions occurred inside of OPC COM callback functions are now caught and logged.
- The values read from the OPC server can now be accessed in on method (SODaCItemIData::setRead) on a per item basis.

### **SOAeC (OPC Alarms and Events Client Library)**

- Exceptions occurred inside of OPC COM callback functions are now caught and logged.

### ***Common components***

#### **OPC Demo Watch**

- With the diagnostic client you can set trace parameters and show the trace messages of the server.
- When receiving a shutdown notification no message box is shown anymore.

#### **OPC Demo Client**

- You can now set some options of the client via the menu Edit|Options.
- The actor id was added to the event list.

#### **Online Help**

- The online help is now integrated into the Microsoft Visual Studio .NET 2002 development environment.