

# SLIK-DA Control

(Simple Language Independent Kit for Creating OPC Data Access Servers)

“Quality OPC Server Implementation Solutions for Software Developers”



## Contents

INTRODUCTION .....	1
BENEFITS & FEATURES .....	2
HOW SLIK-DA WORKS .....	3
IMPLEMENT A SIMPLE OPC DA SERVER—THE BASIC STEPS... 4	
THREE-TIERED DIAGNOSTIC SUPPORT .....	5
PROGRAMMABLE SECURITY INTERFACE .....	7
PRODUCT SUMMARY SHEET .....	8



Complete support for MS.NET. Add OPC DA Server COM interfaces to your C# and VB.NET

## Introduction

The Simple Language Independent Kit for creating OPC Data Access Servers (or SLIK-DA) provides a level of simplicity and flexibility not available before in an OPC server software development toolkit.

SLIK-DA is an ActiveX and .NET Windows Form control used to rapidly implement an OPC DA server using C#, VB.NET, Visual Basic, or Delphi. Just define 6 properties, call 4 methods, implement 2 event handlers, and provide an update method for your server's tags. It can be as simple as that!

Simplicity is not SLIK-DA's only strength. SLIK's flexible programming model places few limits on the design of your server. For example, you can:

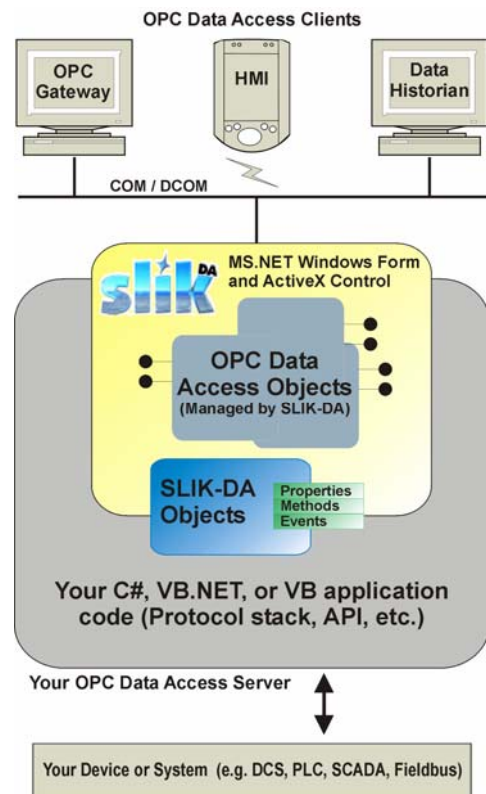
- Build your server's OPC tag collection (your item cache) at startup (static) or as OPC clients make item requests (dynamic),
- Use the built-in OPC item namespace browse support, or implement your own,
- Monitor an OPC client's use of server tags to optimize communications with your device or system.

SLIK-DA is the dependable choice for building your next OPC DA Server. SLIK-DA is built using our field-proven C++ toolkit and includes our standard Three-Tiered Diagnostics Support features.

You can rely on our OPC Experts to provide knowledgeable, responsive, and dedicated support. Get SLIK-DA and our OPC Expert experience working for you on your next project.

## Free Evaluation Software

A free full-featured 30 day evaluation version of SLIK-DA is available to download from our website. The URL is <http://www.OPCexperts.com/SLIKDA>. Then click on Free Evaluation. You can build your own sample server or browse



## Benefits & Features

### Benefits

- ActiveX and MS.NET Windows Form Control for easy implementation.**  
 The SLIK-DA ActiveX implementation means that you can use the toolkit with popular OLE automation-enabled programming tools like Visual Basic, Visual C++, and Delphi. For Visual Studio.NET developers, an engineered Interop Assembly exposes SLIK-DA as a Windows Form Control that can be easily incorporated into your VB.NET and C# projects.
- Cut time from your project schedule.**  
 SLIK-DA will be familiar to programmers working with MS.NET Windows Form and ActiveX controls. With SLIK-DA's built-in features and helpers you can implement a basic OPC Data Access server quicker than ever before.
- Simple for now • Flexible for the future.**  
 Implement a simple server in a few programming steps and retain the flexibility to implement a more complex server in the future, using the same toolkit!
- Manage future software maintenance costs.**  
 As new versions of the data access specification are released, update your server quickly by updating your toolkit.
- Reduce support and integration costs.**  
 SLIK-DA provides a built-in set of tools for diagnosing application, integration, and interoperability problems. Read more about SLIK-DA's Three-Tiered Diagnostic Support later in this document.
- Rely on OPC Expert support.**  
 Northern Dynamic has provided OPC client and server technical support for 6 years to some of the biggest companies in the world. Get our OPC experts and our toolkit behind your next server implementation.

### Features

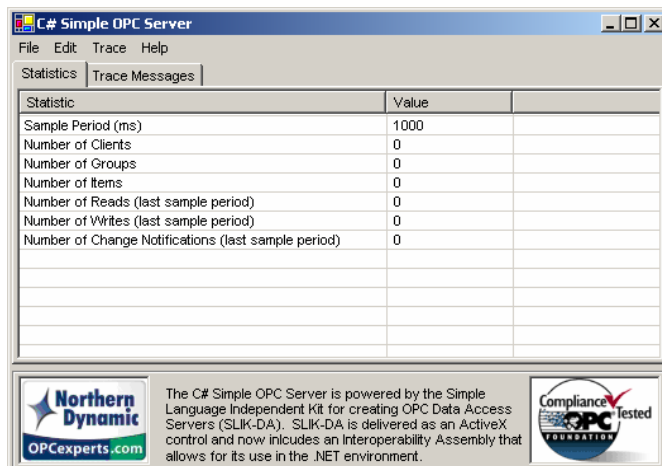
- ActiveX control implementation.
- Includes a custom designed .NET interoperability assembly to ensure efficient and robust MS.NET support.
- All COM and OPC DA specification details are hidden from the developer.
- Simple programming model that is also flexible and the same for all programming languages.
- SLIK-DA utilizes an optimized ActiveX control implementation designed to transfer thousands of items per second.
- Detailed reference manuals included.
- C#, VB.NET, and Visual Basic reference implementations (source code) included.
- Dedicated OPC expert support.
- Extended support and software maintenance programs available.

### OPC Standards Compliant

- OPC Data Access Standard 3.0, 2.0, and 1.0a compliant.
- OPC Security Custom Interface 1.0 compliant.
- Certified compliant with OPC Foundation's Data Access Server Compliance Test Software.

### Built-in Features

- Default implementation for OPC server browse interface provided.
- Default implementation for OPC item validation.
- OPC client cache synchronous and asynchronous read requests serviced by SLIK
- Comprehensive three-tiered diagnostics tools built-in including our new OPC COM Call Tracing technology.



The SLIK-DA evaluation software includes C#, VB.NET, and VB source code for sample server implementations. The servers are certified compliant with the OPC Foundation.

The Simple Server demonstrates SLIK-DA features such as the use of a static tag collection. The server uses the default implementations for browse interface support and OPC item validation.

The purpose of the Custom Server is to demonstrate the flexibility of SLIK-DA. The Custom Server implements a tag collection that is dynamically created as OPC clients make AddItem requests. This server also implements its own browse interface support and OPC item validation.

## How SLIK-DA Works

### Overview

SLIK-DA is a MS.NET Windows Form and ActiveX Control that you use in your MS.NET and OLE automation enabled development environments to create OPC Data Access servers.

Use SLIK-DA to quickly add robust OPC DA interfaces to your existing application. Or, create a new server from the ground up.

SLIK-DA provides a programming model that will be familiar to MS.NET Window Forms Control and ActiveX Control users. The object model, identical for all platforms, is both flexible and scaleable for the future.

**Flexible.** Implement your OPC DA server your way, using as much or as little of SLIK-DA's default implementation as needed.

**Scaleable.** Leverage SLIK-DA's built-in features to quickly implement a robust server. As your server requirements change, override SLIK-DA's default features to optimize your server's design. No need to change your toolkit!

The diagram shown below illustrates how SLIK-DA is used in your OPC server application. The OPC DA server objects (OPCServer, OPCGroup, OPC Item) are automatically created and managed by SLIK-DA. No knowledge of OPC DA or COM is required.

You simply write application code that:

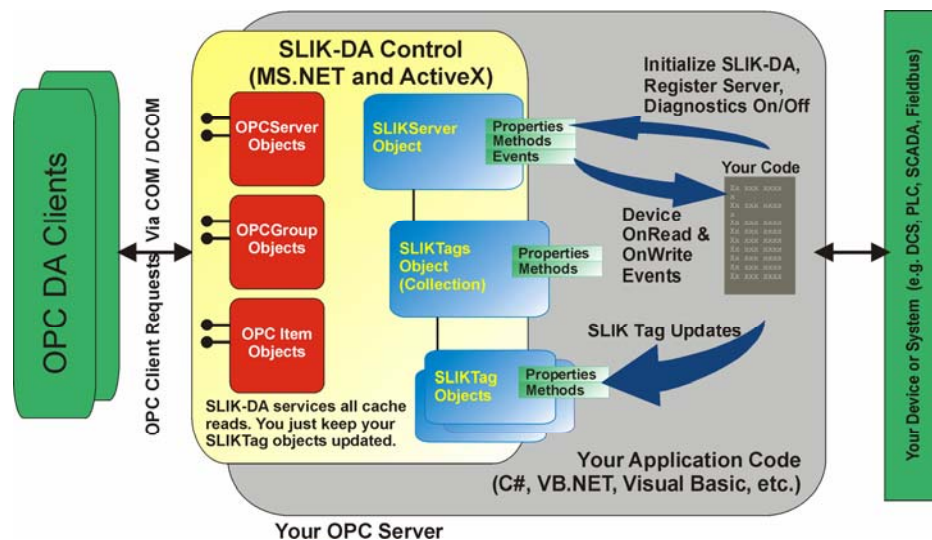
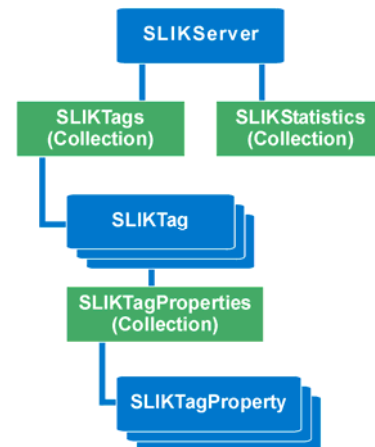
1. Initializes SLIK-DA

2. Provides event handlers for the device read and write operations, and
3. Provides a mechanism for updating the SLIKTag in the SLIKTags collection.

SLIK-DA takes care of OPC client initiated cache read operations and all other OPC Data Access specification and COM details. For MS.NET applications, SLIK-DA handles all .NET and COM interoperability.

### Programming Model

The programmer works with a simple object hierarchy, modifying properties, calling methods, and implementing event handlers. The diagram below shows SLIK-DA's programming object model. For object model details, download the evaluation software and read the SLIK-DA Reference Manual.



## Implement a Simple OPC DA Server – The Basic Steps

### What SLIK Does

Here is a summary of what SLIK-DA does for the programmer.

- Manages all OPC DA client connections for your server.
- For MS.NET users, provides a correct and reliable transition between managed and unmanaged (i.e. OPC/COM) code while maintaining the same programming model.
- Efficiently creates, manages, and destroys OPC DA COM objects as required.
- Transparently manages OPC groups (and contained OPC items) created by an OPC DA client.
- Implements data cache synchronous and asynchronous reads for your application.
- Provides a default implementation for OPC browse interface support. Default browsing is based on the hierarchy of tags in the SLIKTags collection.
- Provides a default implementation for OPC item validation requests.
- Provides flexible server namespace management tools. Statically define your server's namespace at startup or construct the namespace at run-time based on OPC client AddItem requests.
- Provides extensive OPC diagnostic support tools.

### What the Developer Does

As the developer, you work with SLIK-DA object properties, methods, and events to do the following.

- Manage the tags (SLIKTag) in your SLIKTags collection. You can create a static set of tags (e.g. a set of PLC register items) at startup or the SLIKServer

OnAddTag event can inform you when an OPC client has requested an item that is not in your SLIKTags collection (i.e. a dynamic set of tags).

- Implement a method of updating the tags in the SLIKTags collection whenever the underlying value or quality of an item changes in your device or system.
- When requested by SLIKServer event OnRead or OnWrite, perform device reads or device writes.

### The Basic Steps

Follow these basic steps to quickly create an OPC Data Access server using SLIK-DA and VB 6.0.

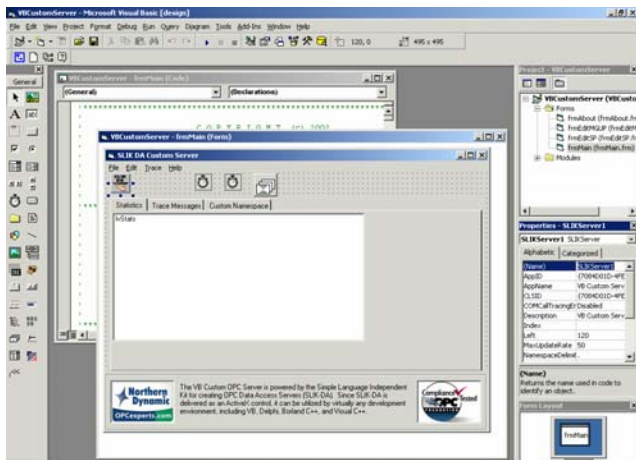
1. Include the SLIK-DA ActiveX control in your project.
2. Define your server's properties using the following SLIKServer object properties: ProgID (e.g. MyCompany.MyServer), AppID (a GUID), and ClassID (a GUID).
3. Process your server's command line parameters **regserver** and **unregserver** using the SLIKServer object's helper methods RegisterServer() and UnregisterServer().
4. Create your collection of tags using SLIKTags collection object method Add().
5. Activate your server's OPC interfaces by calling the SLIKServer object's StartServer() method.
6. Create SLIKServer object event handlers for the OnRead and OnWrite events. OnRead and OnWrite handlers are used to process OPC client initiated device reads and writes.
7. Update the SLIKTag objects in the SLIKTags collection as required by your application.
8. You are done!

Try it yourself! Visit our website to download a step-by-step tutorial and learn how to create your own OPC server.

### Implementation Options

Simplicity is not the only benefit of SLIK-DA. SLIK-DA was designed to be flexible, allowing you to design your OPC server, your way. Briefly, here is a list of additional server features you can implement using SLIK-DA.

- Optimize communications with your device by monitoring for Tag active / inactive state changes using the SLIKServer object's OnTagActiveStateChange event.



## Implementation Options (cont'd)

- Rather than defining a static tag collection for your server, dynamically add and delete SLIKTags from your SLIKTags collection based on OPC client AddItem / RemoveItem requests.
- Override SLIK-DA's default OPC item validation to implement custom item validation logic.
- Manage client connections using the SLIKServer's OnClientConnect event to limit connections or implement software licensing.
- Override SLIK-DA's built-in OPC item browsing to implement your own OPC item browsing.
- Integrate SLIK-DA's Three-Tiered Diagnostic Support features with your application's diagnostic tools.

## Three-Tiered Diagnostic Support

Built into SLIK-DA is a set of diagnostic tools that you and your field support team can leverage in your OPC server deployment. Each tier provides an increasing level of visibility into OPC client interactions with your server.

When faced with field integration issues, you and your support team have at your disposal a set of tools to help quickly and definitively diagnose OPC client-server interoperability problems. This lowers your support costs and helps to identify interoperability problems quickly.

### Diagnostic Tool Overview

The illustration below shows how the diagnostic tools interact with SLIK-DA and your driver code.

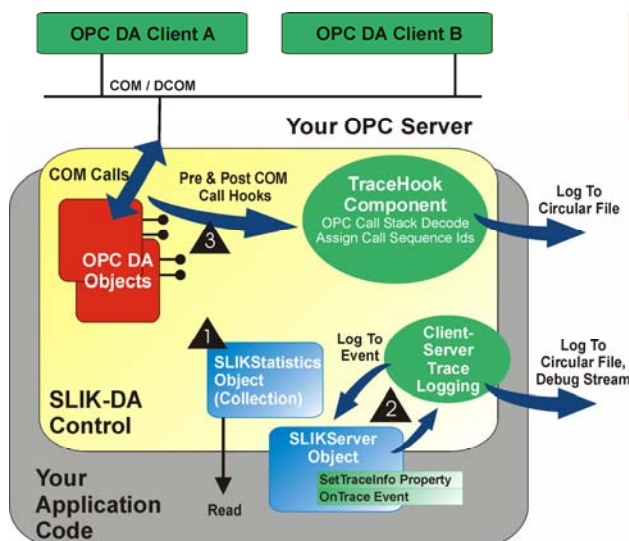
**1 Statistics (Tier 1)** - SLIK-DA automatically tracks various statistics for all client connections. Your driver code can access and log, alarm on, or display these statistics to an end user. The following real-time statistics are available.

- OPC client connection count.
- OPC group count.
- OPC item count.
- Read transactions performed per sample period.
- Write transactions performed per sample period.
- Change notifications performed per sample period.

**2 Client-Server Trace Logging (Tier2)** - This tool generates timestamped messages describing client-server interactions. The easy-to-read, transaction-oriented messages are used by your commissioning and technical support teams to identify common client-server interoperability or system integration problems. When activated, varying levels of detail can be recorded.

- Client connection and server activation events.
- OPC Group create, delete, and modify events.
- OPC Item read, write, and subscription notification events.

Trace messages can be routed to a circular file, to the debug output stream, or integrated with your message logging system by implementing a handler for the SLIK-Server object's OnTrace event. A sample listing is shown in the reference manual (included in the free



This diagram presents a simplified view of the diagnostic features built into SLIK-DA.

SLIK-DA provides a three-tiered set of diagnostic tools to provide visibility into OPC client-server interactions. Use of these tools in the field can reduce your support and integration costs.

Tier 1 - Statistics - client connection statistics.

Tier 2 - Client-Server Trace Logging - detailed listing summarizing transactions.

Tier 3 - OPC COM Call Tracing - thread-specific, exhaustive listing of all OPC COM interface calls.

## Three Tier Diagnostic Support (cont'd)

**▲ OPC COM Call Tracing (Tier 3)** – This component logs timestamped, thread-specific details for all OPC DA interface method calls. Developers use this tool to create an exhaustive and annotated audit trail of OPC interface calls on your server. When activated, the following information is logged for each OPC interface call.

- Pre- or Post-method call timestamp.
- Unique COM object ID for identifying the object instance making the call.
- Unique, sequential call ID for identifying the order of calls on a per thread basis.
- Interface name and method name.
- HRESULT for post-call messages.
- [in] and [out] parameter values.

The following levels of COM call details can be selected for logging:

- Only failed method calls.
- All method calls with HRESULTs only.
- All method calls with HRESULTs only.
- Pre- and post- method call trace messages to

- Pre- and post- method call trace messages with full [in] and [out] parameter dump.

In addition, a filter expression can be specified. You can dynamically filter the message stream, reducing the amount of information logged to that which is of interest.

OPC COM Call Tracing is implemented by the NDI TraceHook Component, a DLL that hooks into OPC COM calls to capture and record the call stack. Call stack information is decoded, annotated, and then logged to a circular file.

Since the TraceHook component hooks the COM calls (and does not operate in a pass-through mode) OPC COM Call Tracing is very efficient, imposing very little run-time overhead. Therefore, timing related integration problems can be diagnosed.

The TraceHook component ships with a simple utility application that allows tracing parameters to be changed dynamically.

OPC COM Call Tracing has been included with the sample servers (installed with the SLIK-DA trial software). Just enable the COM Call Tracing option and connect to the simulation server using your OPC client.

### Excerpts from a Sample OPC Com Call Trace Listing

This listing illustrates the OPC COM Call interactions between an OPC client browsing the namespace of the Customer Server reference implementation installed with the SLIK-DA evaluation software.

```
2001-07-31 12:58:51.093 LOG FILE OPENED!
12:58:51.213 VbCustomServer:OPCServer@bf1450 >> (207-1) IOPCServer::AddGroup("NDI_Browser", 0, 0, 0, [Ptr] NULL, 0, 2048, [out], [out], REC<968964692, 286, 4560, >, [out])
12:58:51.233 VbCustomServer:OPCServer@bf1450 << (207-1) hr=0x4000d IOPCServer::AddGroup([in], [in], [in], [in], [in], [in], [in], 12529312, 10, [in], [Unknown] 0x00148E38)
12:58:51.283 VbCustomServer:OPCServer@bf1450 >> (311-2) IOPCBrowseServerAddressSpace::QueryOrganization([out])
12:58:51.283 VbCustomServer:OPCServer@bf1450 << (311-2) hr=0 IOPCBrowseServerAddressSpace::QueryOrganization(ENUM<1>)
12:58:51.283 VbCustomServer:OPCServer@bf1450 >> (207-3) IOPCBrowseServerAddressSpace::BrowseOPCItemIDs(ENUM<1>, "", 0, 0, [out])
12:58:51.303 VbCustomServer:OPCServer@bf1450 << (207-3) hr=0 IOPCBrowseServerAddressSpace::BrowseOPCItemIDs([in], [in], [in], [in], [Unknown] 0x015F0180)
12:58:51.303 VbCustomServer:OPCServer@bf1450 >> (311-4) IEnumString::RemoteNext(1, [out], [out])
12:58:51.303 VbCustomServer:OPCServer@bf1450 << (311-4) hr=0 IEnumString::RemoteNext([in], "src", 1)
12:58:51.303 VbCustomServer:OPCServer@bf1450 >> (333-5) IOPCBrowseServerAddressSpace::GetItemID("src", [out])
12:58:51.303 VbCustomServer:OPCServer@bf1450 << (333-5) hr=0 IOPCBrowseServerAddressSpace::GetItemID([in], "src")
...
12:58:53.066 VbCustomServer:OPCGroup$NDI_Browser >> (146-17) IOPCItemMgt::ValidateItems(1,
    ARRAY<
        ELEM[0] REC<"", "src.Tag01", 0, 0, 0, [Ptr] 0x00000000, 0, 0>
    >, 0, [out], [out])
12:58:53.066 VbCustomServer:OPCGroup$NDI_Browser << (146-17) hr=0 IOPCItemMgt::ValidateItems([in], [in], [in],
    ARRAY<
        ELEM[0] REC<0, 11, 47789, 1, 0, [Ptr] 0x00000000>
    >,
    ARRAY<
        ELEM[0] 0
    >)
12:58:53.066 VbCustomServer:OPCServer@bf1450 >> (311-18) IOPCBrowseServerAddressSpace::BrowseAccessPaths("src.Tag01", [out])
12:58:53.086 VbCustomServer:OPCServer@bf1450 << (311-18) hr=0 IOPCBrowseServerAddressSpace::BrowseAccessPaths([in], [Unknown] 0x015F0180)
...
12:58:57.652 VbCustomServer:OPCServer@bf1450 >> (333-1216) IOPCServer::RemoveGroup(12529312, 1)
12:58:57.702 VbCustomServer:OPCServer@bf1450 << (333-1216) hr=0 IOPCServer::RemoveGroup([in], [in])
2001-07-31 12:59:01.007 LOG FILE CLOSED!
```

## Programmable Security Interface

### Overview

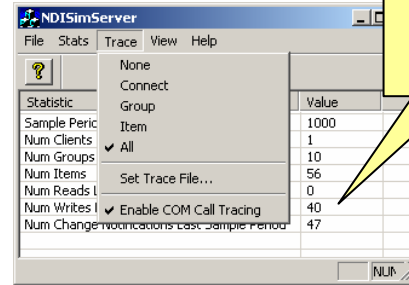
Protecting sensitive data located on your OPC server can be done quickly and easily using the OPC Private Security Interface provided with this toolkit.

### Benefits of Our OPC Security Implementation

The SLIK-DA Toolkit now provides a complete implementation of the OPC Private Security Interface that goes beyond providing simple interface hooks. It ships with the following features and a sample source code designed to make your job easier.

- Client Logon — With OPC private security, clients must explicitly logon before being allowed access to any secured object.
- Unique Security Credentials — Private security credentials allow you to isolate server security from underlying network security.
- Security Granularity — Two levels of security granularity give you added flexibility and control. Server level security allows you to define user access rights for all items in the server namespace. Item level granularity allows you to assign user access rights to individual items or to groups of items.

- High Performance — Our OPC Private Security implementation imposes virtually no overhead to normal OPC server processing.
- Rapid Implementation — Modify the sample server (provided with our evaluation software) to get your security implementation working quickly.
- Security Interface Testing — Our SLIK-DAC product (for building data access clients) has security features added as well. Use this toolkit to develop a client that can effectively test your server security implementation.
- OPC Standard Compliant — The SLIK-DA Toolkit security interface and sample implementation are fully-compliant with the OPC Security Custom Interface 1.0 Standard.



A simulation server (including source code) is installed with the demo software.

Object	Interface	OPC Version	Supported	
OPCServer	IOPCCommon	2.0, 3.0	Yes	
	IOPCServer	1.0, 2.0, 3.0	Yes	
	IOPCServerPublicGroups (optional)	1.0, 2.0	No	
	IOPCBrowseServerAddressSpace (optional)	1.0, 2.0	Yes	
	IOPCBrowse	3.0	Yes	
	IOPCItemProperties	2.0	Yes	
	IConnectionPointContainer	2.0, 3.0	Yes	
	IPersistFile (optional)	1.0, 2.0	No	
	IOPCItemIO	3.0	Yes	
	IOPCSecurityPrivate	1.0	Yes	
	OPCGroup	IOPCGroupStateMgt	1.0, 2.0, 3.0	Yes
		IOPCGroupStateMgt2	3.0	Yes
		IOPCPublicGroupStateMgt (optional)	1.0, 2.0	No
		IOPCItemMgt	1.0, 2.0, 3.0	Yes
IOPCItemDeadbandMgt		3.0	Yes	
IOPCItemSamplingMgt (optional)		3.0	No	
IOPCSyncIO		1.0, 2.0, 3.0	Yes	
IOPCSyncIO2		3.0	Yes	
IOPCAsyncIO		1.0	Yes	
IOPCAsyncIO2		2.0, 3.0	Yes	
IOPCAsyncIO3	3.0	Yes		
IConnectionPointContainer	IConnectionPointContainer	2.0, 3.0	Yes	
	IDataObject	1.0	Yes	
EnumOPCItemAttributes	IEnumOPCItemAttributes	1.0, 2.0, 3.0	Yes	
	IOPCSecurityPrivate	1.0	Yes	

## Product Summary Sheet

### System Requirements

<b>Operating Systems</b>	Windows 98, Windows NT 4.0 (SP5 or later), Windows 2000, Windows XP, Windows Server, Windows Vista
<b>OPC Clients Supported</b>	All that are compliant with OPC Data Access 1.0a, 2.0, and 3.0 standards.
<b>Development Environment</b>	- MS Visual Studio.NET - All OLE automation enabled tools like Visual Basic and Delphi

### Product Licensing

A SLIK-DA license is required for each development computer using the ActiveX control to create OPC servers. The OPC server you develop will only run if the server executable is generated on a SLIK-DA licensed computer. Re-distributables can be used in any number of products and no royalty fees are required. Annual developer support contracts are available.

### Product Support

Northern Dynamic's dedicated support team is available via telephone and e-mail.

Your new license purchase includes 30 days of unlimited product support via email and telephone. The 30 days start from the date of first contact. The 30 day support period expires if not used within 180 days from the date of purchase.

Northern Dynamic's OPC experts are ready to assist with the development of your OPC server. Comprehensive on-site support and best-practices training are available. Contact a sales representative today for more information.

### Proven Interoperability

SLIK-DA offers guaranteed compliance with the OPC Data Access standard. The reference implementation servers included with SLIK-DA are certified compliant with the OPC Foundation's Data Access Compliance Test Software.

### Ordering Information

You can try SLIK-DA for free by downloading a evaluation version. When you are ready to purchase, contact Northern Dynamic to obtain your license. Your evaluation software can be converted to a licensed version immediately.

Contact Northern Dynamic for pricing information and a quotation.

telephone	+1 (519) 725-2071
	+1 (888) 265-7345 (toll free in NA)
fax	+1 (519) 725-2072
email	solutions@nordyn.com
website	http://www.OPCexperts.com

### About Northern Dynamic Inc.

Our team of OPC experts can help you embed OPC technology into your software products, supply robust integration solutions, and provide third party support during your integration efforts.

Visit our website at [www.OPCexperts.com](http://www.OPCexperts.com) to learn more about our OPC solutions.

- OPC Toolkits · OPC Gateway (for server data exchange)
- OPC Consulting · OPC Software Development
- OPC Technical Support



Northern Dynamic Inc.  
Suite 3—295 Hagey Blvd  
Waterloo Research & Technology Park  
Waterloo, Ontario  
Canada N2L 6R5  
Tel : +1 (519) 725-2071  
Fax : +1 (519) 725-2072

