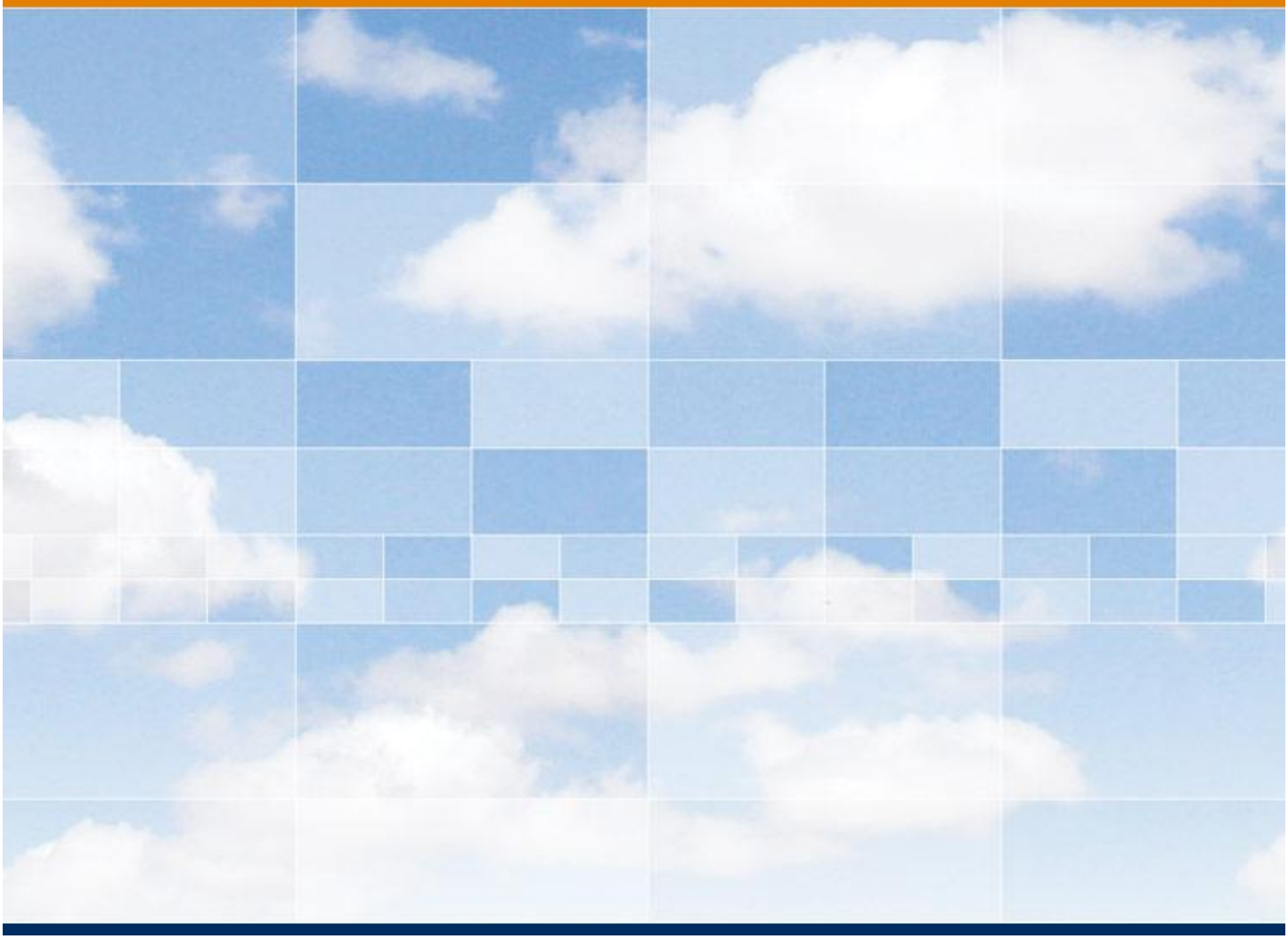
MatrikonOPC



MatrikonOPC Tunneller

User's Manual

MatrikonOPC Tunneller
User's Manual

This manual is a product of Matrikon Inc.

Matrikon Inc.
Suite 1800, 10405 Jasper Avenue
Edmonton, AB T5J 3N4
Canada

Phone: +1.780.448.1010
Fax: +1.780.448.9191
www.matrikonopc.com

Document Revision History:

Date	Document Version	Description	Author
2004-21-21	1.0	Initial document.	RT
2005-07-28 2006-03-09	1.1 – 1.4	Updated to version 2.2, added clarification, added HDA, minor revisions.	RS
2006-11-07	1.5	Updated and edited.	TNM
2007-03-14	1.6	Updated and edited.	RN
2007-04-23	2.0	Converted to new template, general edit.	LB
2007-11-19	2.1	Updates to Installation, Troubleshooting sections re: using Stratus box.	RN
2007-11-19	2.2	Updated for software version 3.1.0.0; added Analyzer install note to Installation section.	RN, LB
2008-02-27 2008-03-17 2008-03-27	2.3 – 2.5	Additional software version 3.1.0.0 changes: updates to Remote Tunneller Connection section, updates to Encryption, Compression, User Impersonation and Restriction of the Access to OPC Servers section. Added new section – Advanced SSC Settings. New installer/un-installer, installed files updated.	RN, LB
2008-08-14 2008-08-15 2009-01-08	2.6 – 2.8	Updated to software version 3.1.2.0, updates to OPC Servers Access Restriction and MatrikonOPC Gateway Integration. Updates to Software Requirements and Troubleshooting sections. Updates to Troubleshooting section.	RN, LB

2009-03-11	3.0	Updated to software version 3.2.0.0. Added the Handling Shutdown Request from Remote OPC Server section. Updated the following sections: Licensing, GetStatus Call, User Impersonation, Advanced SSC Settings, OPC Server Access Restriction and MatrikonOPC Security Gateway Integration, Connection and Reconnection, Troubleshooting.	LB, RN
2009-04-23	3.1	Troubleshooting section updated.	RN
2009-11-09	4.0	Updated software version to 3.2.3.0. Removed DEP warning. Updated DCOM hyperlink. Formatting fixes. Contacting Support section updated.	SL, LB
2009-11-09	4.1	Added Limitations section. Updated Installation section.	LB
2009-11-09	4.2	Software Requirements updated.	LB
2009-11-11	4.3	Removed Product Registration screen and related procedural steps from Installation section.	LB
2010-02-18	5.0	Updated document to software v3.2.4.0.	LB
2010-03-17	5.1	Updated Software Requirements, Installed Files, and OPC Compliance sections. Added Get Status Ping description to Table 15. Replaced screenshot (Figure 31) to include new field.	SN, LB
2010-04-08	6.0	Updated software version to 3.2.5.0 Updated the Software Requirements, Installed Files, OPC Compliance, and User Impersonation sections.	SN, ZA, LB
2010-06-01	6.1	Beta tag lines removed. Updated Contacting Support section.	SN, LB
2010-06-24	7.0	Ported to TFS. Software updated to v3.3.0. Manual converted to standard template.	LB
2010-08-30	7.1	Updated Appendix C – Installed Files.	LB
2010-09-08	7.2	Add Remote Tunneller Connection screenshot updated to Communication Timeout field default value updated from 3 seconds to 5 seconds. Communication Timeout field description updated to reflect same.	LW, LB
2010-10-05	7.3	Trademark Information and Introduction sections updated.	LB
2011-01-21	8.0	Updated software version to 3.5.0. Software Requirements updated to included Microsoft Windows Server 2008. New sub-section added to Connection and Reconnection section.	SN, LB
2011-02-25	9.0	Updated software version to 3.5.1.	LB

2011-05-03	10.0	Updated software version to 4.0.0. The following sections were updated to reflect that A&E is now supported: Remote Tunneller Connection, Connection Failure Scenario, OPC Compliance, Appendix C – Installed Files.	MJL, LB
2011-05-16	10.1	Updated the following sections to reflect A&E Support changes: References, Connection Failure Scenario, Limitations, Troubleshooting, OPC Compliance.	CGAP, LB
2011-05-31	10.2	Updated Software Requirements and OPC Compliance sections.	GEAK, LB
2011-06-03	10.3	Figure 5 replaced to show current MatrikonOPC marketing scheme in bottom screen section.	LB
2011-11-03	11.0	Updated software version to 4.0.1. Replaced Figure 21 screenshot to include Commit All Log File Writes checkbox. Checkbox description added to Table 8.	LB
2011-12-16	12.0	Updated software version to 4.0.2.	ISY
2012-03-05	13.0	Updated software version to 4.0.3. Updated Copyright Information and Contacting Support sections.	LB
2012-04-30	14.0	Updated software version to 4.0.4.	LB
2012-06-05	15.0	Updated software version to 4.0.5. Added Pi usage item to Troubleshooting section.	LB
2012-09-17	16.0	Updated software version to 4.1.0.	LB
2012-10-15	16.1	Updated Figure 3 and Table 2 in Remote Tunneller Connection section. Updated Limitations and Troubleshooting sections, and Appendix B - Installation.	CGAP, LB
2012-10-22	16.2	Updated Tables 2, 3, and 8, Figure 21, and Appendix C – Installed Files. Typos fixed.	LB

SOFTWARE VERSION

Version: 4.1.0

DOCUMENT VERSION

Version: 16.2

COPYRIGHT INFORMATION

CONFIDENTIAL

IMPORTANT NOTICE

LICENSE AGREEMENT

TRADEMARK INFORMATION

OTHER

MatrikonOPC™ is a division of Matrikon™ Inc.

Table of Contents

Table of Appendices

Table of Figures

Table of Tables

Introduction

Companies wishing to use OPC technology to link operators and engineers with plant devices, often encounter communication problems. The majority of these problems occur not during normal operation, but at the time of installation. Quite often plant engineers face difficulties configuring cross-network communication, windows authentication, as well as start-up and run-time permissions.

MatrikonOPC™ Tunneller alleviates many of these problems by providing a mechanism for OPC data communication without the use of distributed COM (DCOM). Tunneller provides the following:

- Cross-domain and cross-workgroup communication with minimal network configuration.
- Bypassing of Microsoft Windows network authentication used by DCOM.
- A finer level of control over communication timeouts.

By eliminating common DCOM hurdles, Tunneller enables the smoothest possible installation and operation of OPC technology in any environment.

Who Should Use This Manual

This manual is intended for use by all users of MatrikonOPC Tunneller.

This manual explains how to install and configure the software, and how to perform common tasks. In addition, technical information about OPC data items is included, along with sections on diagnostics and troubleshooting.

Overview of Manual

This document uses icons to highlight valuable information. Remember these icons and what they mean, as they will assist you throughout the manual.

	This symbol denotes important information that must be acknowledged. Failure to do so may result in the software not functioning properly.
BOLD	Font displayed in this color and style indicates a hyperlink to the applicable/associated information within this document, or if applicable, any external sources.

The *User's Manual* has been designed as such so that you can click on references in the document to jump to that referenced point without having to scroll through several pages (in some cases). For example, if you were to see the sentence "*Refer to Figure 1 for more information*", pressing the **CTRL** key and clicking your mouse on the text "*Figure 1*" automatically takes you to the location of Figure 1 within the document.

This manual consists of several sections and is structured as follows:

- **Introduction** – this introductory chapter.
- **Getting Started** – provides system requirements information.
- **Configuration** – shows how to configure Tunneller, and describes each component in detail, including windows/screens, panels, tabs, and menu commands.
- **Encryption, Compression, User Impersonation, and Restriction of the Access to OPC Servers –** shows how to use Tunneller's Encryption and Compression features.

- **Connection and Reconnection** – presents a simple scenario demonstrating how Tunneller reacts to an interrupted network.
- **Tunneller with MatrikonOPC Redundancy Broker** – provides useful information about using Tunneller with Redundancy Broker (ORB).
- **Limitations** – provides information on specific performance and operational limitations of the software.
- **Troubleshooting** – provides licensing, MatrikonOPC Support contact information, solutions for common problems that may be encountered, and answers to frequently asked questions.
- **OPC Compliance** – details supported interfaces with regard to installation, common interfaces, and data access.
- **Appendices**:
  - **A** – Standard Data Types
  - **B** – Installation
  - **C** – Installed Files
  - **D** – Un-Installation

References

This document references information found within the following documents/sites:

- **www.opcfoundation.org**
- **www.matrikonopc.com**
- **www.opcsupport.com**
- *OPC Overview 1.0*
- *OPC Common Definitions and Interfaces 1.0*
- *OPC Data Access Specification 2.05a*
- *OPC Historical Data Access Specification 1.20*
- *OPC Alarms and Events Specification 1.10*

Document Terminology

The following terms are used interchangeably throughout this document:

- *screen* and *window*
- *tab* and *panel*

Table 1 provides a list of definitions for terms used throughout this document.

Term/Abbreviation	Description
A&E	OPC Alarms and Events. Provides access to process alarm and event data.
CCT	Tunneller Client Configuration Tool.
COM	Component Object Model. A method for organizing software, specifying how to build components that can be dynamically interchanged.
CS	In short form used to indicate the Client-Side machine.

Term/Abbreviation	Description
CSC	Tunneller Client-Side Component (or Client-Side Gateway).
DA	OPC Data Access. Provides access to real-time process data.
DCOM	Distributed Component Object Model. An extension of COM that allows communication between COM components over a network.
DDE	Dynamic Data Exchange. Allows the transfer of data between two running applications.
HDA	OPC Historical Data Access. Provides access to historical process data.
HMI	Human Machine Interface. Device that allows interaction between the user and machine. Typically used in process control applications.
Matrikon	Matrikon Inc.
MatrikonOPC	Matrikon's brand name for its OPC servers and clients.
OPC	A communication standard. Refer to **www.opcfoundation.org** for more information.
ORB	MatrikonOPC Redundancy Broker.
PLC	Programmable Logic Controller.
SS	In short form used to indicate the Server-Side machine.
SSC	Tunneller Server-Side Component (or Server-Side Gateway).

Table 1 - Terms and Definitions

Getting Started

This chapter contains important information about installing Tunneller and how to contact the MatrikonOPC Support team.

The **System Requirements** section shows how to avoid future problems by ensuring that the system meets the minimum software and hardware requirements. Detailed step-by-step instructions in **Appendix B - Installation** walks you through the installation process. **Appendix C – Installed Files** lists the files that are installed during this process.

Once the software is installed, refer to the **Licensing** section for information on how to obtain the appropriate license. The *Licensing* section will refer you to the *Licensing Procedures* document that was installed along with the server and this *User's Manual*. If any problems are encountered during installation or licensing, refer to the **Contacting Support** section for information about how to contact the MatrikonOPC Support team for assistance.

System Requirements

The software has minimum **Software** and **Hardware** system requirements. These requirements must be met for the software to function properly.

Note: To install and configure a MatrikonOPC server, you must be set up as an administrative user account rather than a restricted user account.

Software Requirements

The server requires the following software:

- Microsoft Windows 2000 Server SP4, or
- Microsoft Windows XP SP2, or
- Microsoft Windows 2003 SP0, or
- Microsoft Windows 7, or
- Microsoft Windows Server 2008 SP0
- Microsoft .NET 2.0 Framework (included with this install program)

Note: It is recommended that the most current service packs are installed.

Hardware Requirements

The server requires the following hardware:

- Intel® Pentium® 4 Processor
- 512 MB RAM
- 40 GB 7200 RPM Hard Drive
- TCP/IP connectivity

Configuration

The goal of Tunneller is to provide trouble-free communication in a manner as transparent as possible to an existing OPC installation. Tunneller achieves this by acting as a pass-through between the OPC client and the OPC server (see Figure 1).

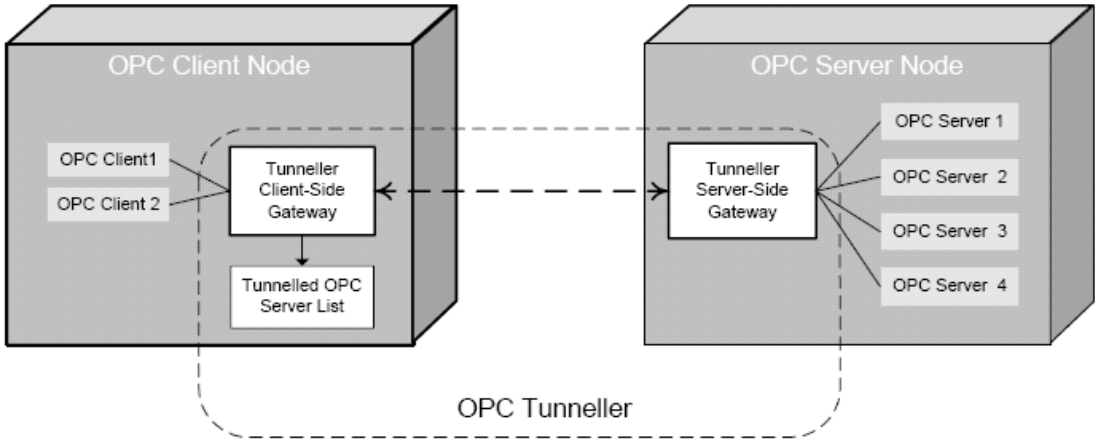


Figure 1 - High-Level Schematic of Tunneller Configuration

Main Screen

Basic Tunneller configuration is done with the MatrikonOPC Tunneller Client-Side Gateway Configuration Utility, which is available on any computer where the CSC files have been installed.

Click on Windows **Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller**, and open the **Client-Side Gateway Configuration Utility** (Figure 2). This is also known as the **Client-Side Configuration Tool**, or CCT.

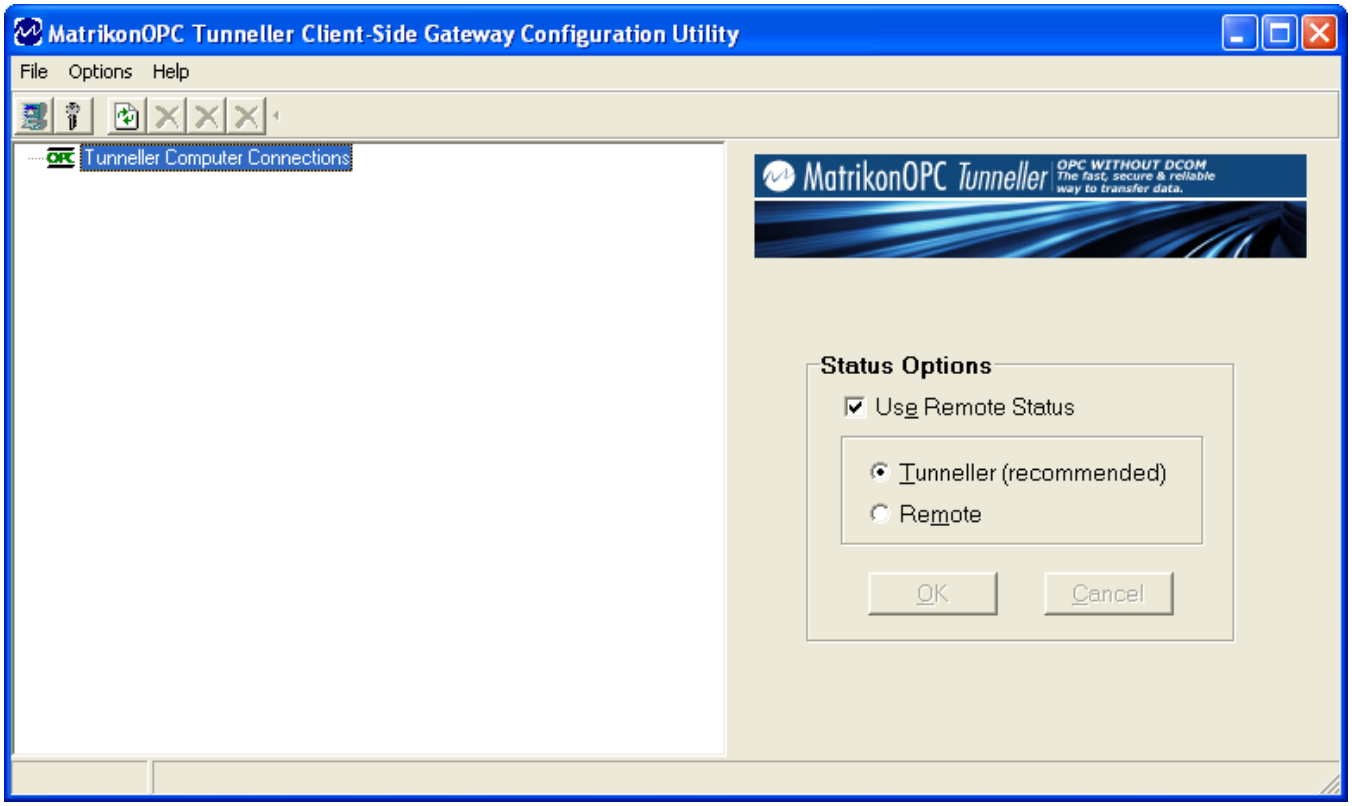


Figure 2 - MatrikonOPC Tunneller Client-Side Gateway Configuration Utility

This screen contains all Tunneller connections that have been added. These are connections to OPC servers located on machines where the Server-Side Tunneller files have been installed. If there are no Tunneller connections configured, then the **Client-Side Gateway Configuration Utility** will display no available Tunneller connections.

Tunneller connections may now be added.

To add a connection:

1. From the **Client-Side Gateway Configuration Utility** screen, either
   - Press **Ctrl+N**, or
   - Click on the leftmost button on the toolbar (the computer icon ), or
   - Go to the **File** menu and select the **Add Remote Tunneller Connection** option.

Note: For Tunneller to operate, at least one connection to an OPC Server must be configured (local connections are also allowed). This means that Tunneller must be installed on a PC that is accessible via TCP/IP.

2. The **Add Remote Tunneller Connection** window (Figure 3) appears.

Remote Tunneller Connection

The **Add Remote Tunneller Connection** window (Figure 3) contains all of the information necessary to set up a Tunneller CSC connection to a remote PC with the Tunneller SSC installed. The OPC server to which Tunneller will connect must reside on the same machine as the remote Tunneller install.

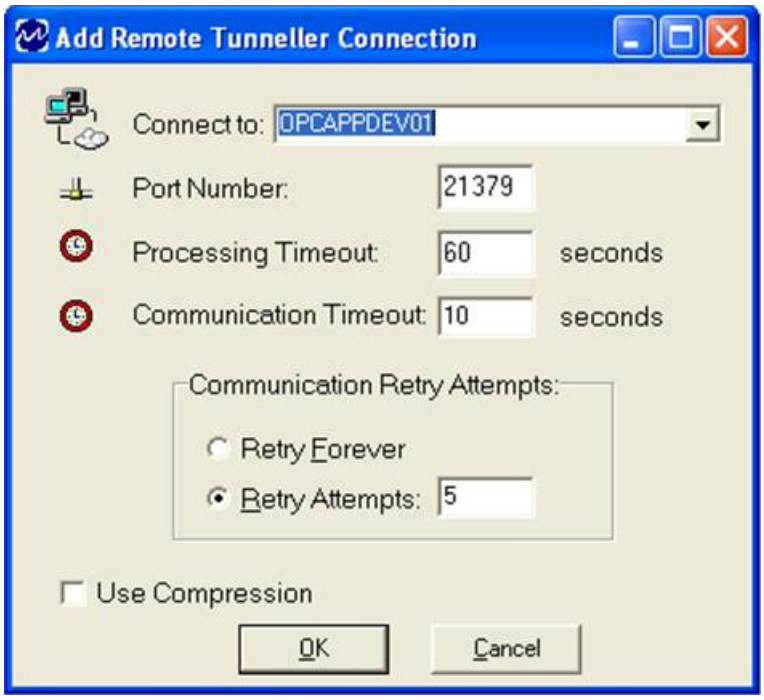


Figure 3 - Add Remote Tunneller Connection Window

Table 2 describes the fields in the **Add Remote Tunneller Connection** window (Figure 3).

Option	Description
Connect to	Allows you to enter or select (from the drop-down list) the name of the computer referenced either by its IP address or by the specific computer name.
Port Number	Allows you to enter the port number on the remote PC to which Tunneller should establish a connection. Restarting the Tunneller service is required when this option is changed. The default port is ***21379***. **Note:** To successfully communicate, the port number configured on the Client-Side Component must match with the port number configured on the Server-Side Component.
Processing Timeout	Allows you to enter the amount of time (in seconds) the CSC waits for a response to a request before it considers the request failed and returns an error code to OPC client. **Processing Timeout** should be set to a value higher than the longest time required to execute a request. The default value is ***60*** seconds.
Communication Timeout	Allows you to enter the amount of time (in seconds) the network communication mechanism will try to send a request or receive a response from the SSC before it considers the communication failed. **Communication Timeout** should be configured to a value higher than the time required to send the largest message over the network. The default value is ***10*** seconds.
Communication Retry	Allows you to select an option to specify whether the failed

Option	Description
Attempts	communication is to be retried forever (***Retry Forever***), or only for a specified number of attempts (***Retry Attempts***). If the Retry Attempts option button is selected, in the adjacent field, enter the number of times Tunneller should try to resend requests/responses that have failed to send. This parameter defines how many attempts should be made after failure. The default value is ***5*** (i.e., sending each request/response will be attempted up to six times).
Use Compression	Enable compression for remote Tunneller connections. By default this checkbox is not selected. **Note:** Compression can be adjusted later for each individual OPC server.

Table 2 - Add Remote Tunneller Connection Options

Note: If Tunneller detects that the network path to the remote computer is unavailable because of a physical break in the network or the remote SSC is not available, it considers the sending of the command as failed and will not attempt any retries. At that point, Tunneller will try to reconnect to the remote computer.

Once the settings have been configured to the user's specific network requirements, click on the **OK** button. If the specified port number is open and available, the SSC provides a list of installed OPC servers to the CSC. The **Client-Side Gateway Configuration Utility** (Figure 4) will now display the list of accessible (through Tunneller) OPC servers for the selected remote host.

If the OPC server supports more than one OPC interface (i.e., A&E, DA, or HDA), then the list will contain one Tunneller entry for each interface for that OPC server.

Note: In version 3.1.0.0 of Tunneller, new functionality is added to restrict access to OPC servers. If this feature is turned on, then the SSC can return a smaller, or empty, list of OPC servers.

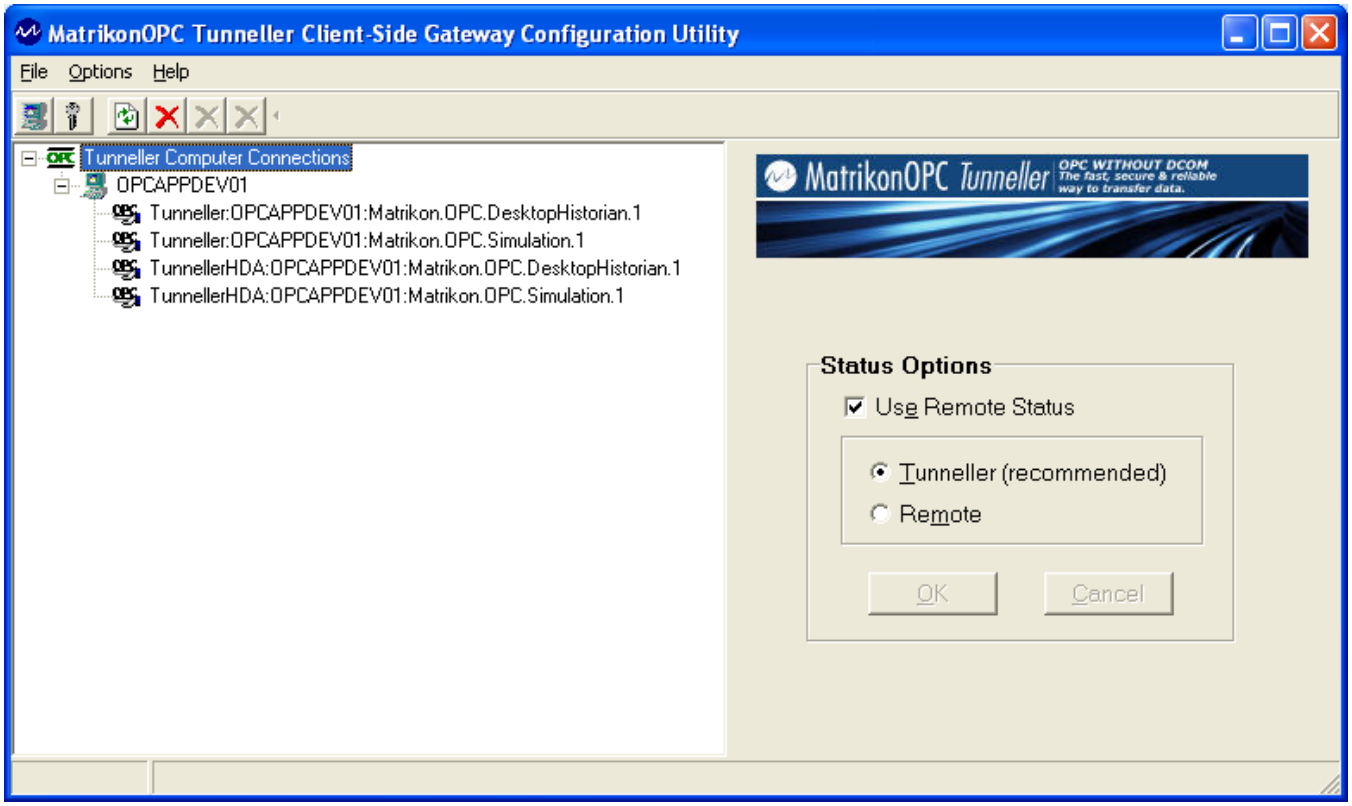


Figure 4 - Tunneller Server Configuration

When the **Tunneller Configuration Utility** shows a list of tunnelled OPC servers, OPC clients may browse for and connect to those OPC servers. Figure 5 shows **MatrikonOPC Explorer** browsing for and finding Tunneller DA OPC servers.

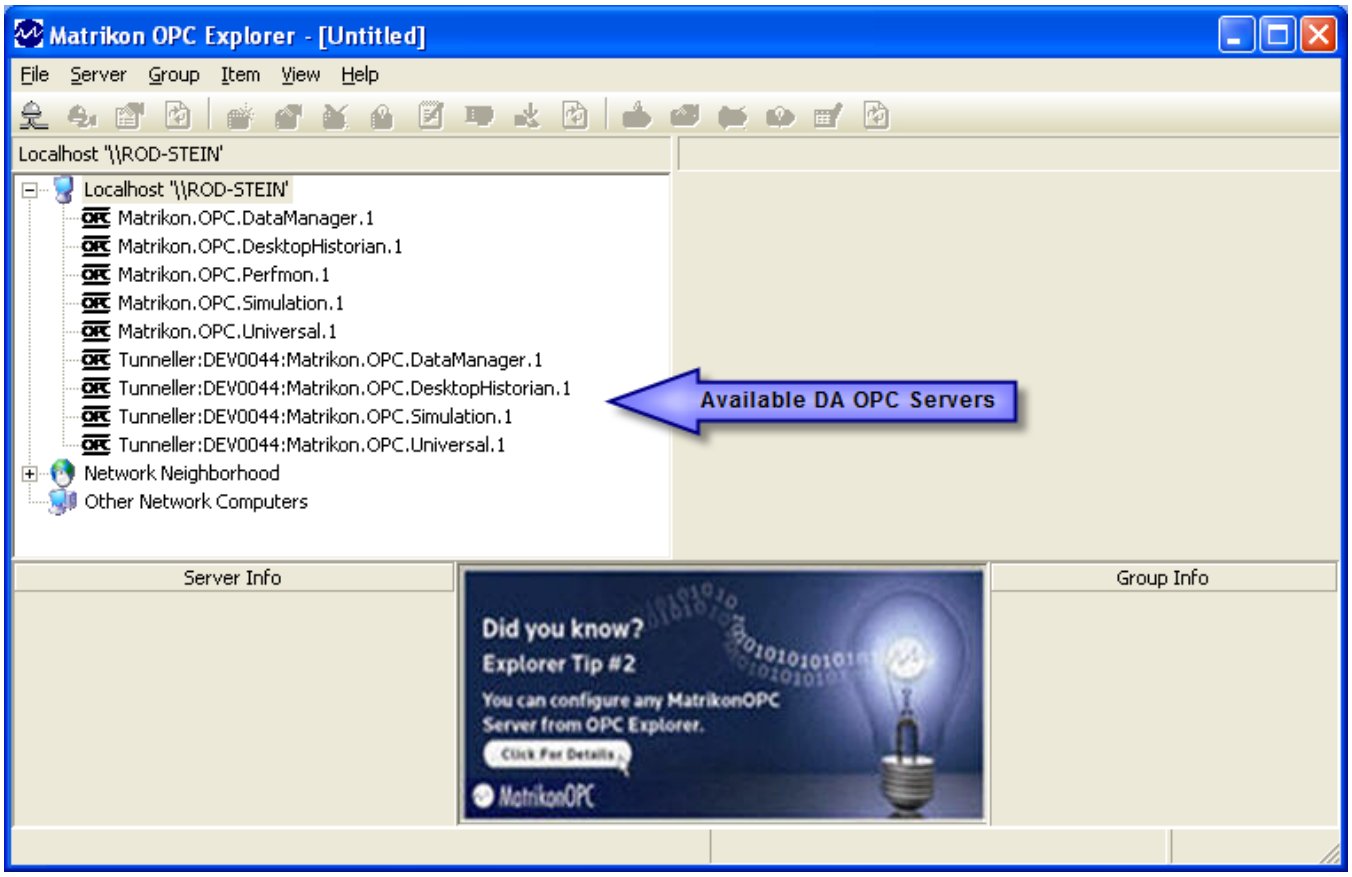


Figure 5 - MatrikonOPC Explorer: Available Connections to Tunnelled OPC Servers

Options Menu

The **Options** menu (see Figure 6) provides several functions. One such function is to refresh settings for all tunnelled servers by reading them from the system registry. This does not affect any current connections and does not create connections to Server-Side components — it only refreshes the configuration display.

To refresh the server list:

1. Click on **Options** -> **Refresh Server List**, or

   Press **F5**.

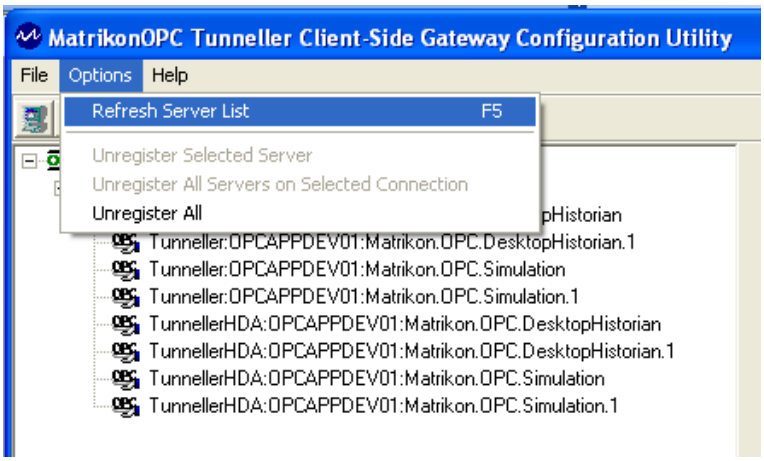


Figure 6 - Options Menu List

The user can un-register certain Tunneller server connections from the toolbar. Click on **Options** -> **Unregister Selected Server**, or press the **black X** (Figure 7), to un-register the server currently selected in the Client Configuration Tool.

Figure 7 - Remove Selected Tunneller Connection

Click on **Options** -> **Unregister All Servers on Selected Connection**, or press the **blue X** (Figure 8), to un-register all connections to the computer selected (e.g., ***OPCAPPDEV01*** in Figure 4) in the Client Configuration Tool.

Figure 8 - Remove All Connections to Selected Computer

Click on **Options** -> **Unregister All**, or press the **red X** (Figure 9), to un-register all configured Tunneller connections.

Figure 9 - Remove All Tunneller Connections

WARNINGS:

- Before removing or modifying any Tunneller server connection, make sure that no clients are actively connected. If there are active connections, the OPC clients must be disconnected before the Tunneller server connection is removed.
- When removing a Tunneller server connection and recreating it, ensure that any OPC client that has the connection information cached, clears its cache and obtains the new Tunneller server connection information. **Failure to do so may cause unpredictable behaviour.**

OPC Client Requirements

The OPC clients connecting to Tunneller are required to fulfill the following requirements:

- Allow in proc servers. A small portion of Tunneller will be loaded in proc.

Note: Starting in version 3.1.0.0, Tunneller CSC can also be loaded as a remote COM server (i.e., using **CoCreateInstanceEx** function call).

- Support the Shutdown call-back. This is a required interface but some clients do not support it. It must be supported for Tunneller to relay communication failure information.

GetStatus Call

Tunneller uses TCP connections to do its data transfer, but does not automatically detect TCP line failures if no requests are made by the OPC client. The **GetStatus** call should be used as a heartbeat mechanism for line failure detection. The OPC client must call **GetStatus** at regular intervals to ensure the line is available.

Tunneller does not initiate the **GetStatus** call on its own. It relies on the OPC client to make the **GetStatus** call. The call can be made as often as needed, depending on the required detection level.

Frequent calls (at approximately one second) will detect a line failure quickly. It will also detect a line that has gone down but has come back immediately. Infrequent calls can allow a momentary interruption to pass but will not detect a prolonged failure until the line is needed by an actual call. The user's specific requirements will indicate how often the **GetStatus** should be called.

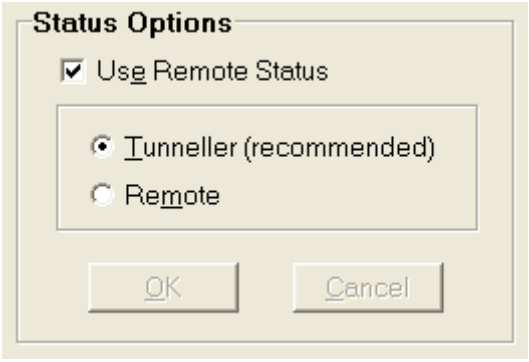


Figure 10 - GetStatus Settings

Figure 10 shows the configuration options for the **GetStatus** call. Options are common for all tunnelled OPC servers and available to configure when the root of the tree control (***Tunneller Computer Connections***) is selected on the **Client-Side Gateway Configuration Tool**. These options are described in Table 3.

Option	Description
Use Remote Status	Selecting this checkbox will cause the **GetStatus** call to be made across the TCP line to the end OPC server. Clearing the checkbox will cause any **GetStatus** call made to Tunneller to be returned immediately with Tunneller's own status.
Tunneller (recommended)	**Recommended by MatrikonOPC.** This option is available only when the **Use Remote Status** checkbox has been selected. When selected, this option forces an internally generated **GetStatus** message to be sent to the end OPC server. However, it returns Tunneller's own status immediately to the calling client, so it is a "non-blocking call". The remote call will be made as close as possible to the rate at which the end OPC client is calling **GetStatus** on Tunneller. If a communication failure is detected, Tunneller's status will be set to failed. In subsequent **GetStatus** calls, the status field of the status structure

Option	Description
	returned by Tunneller will be set to failed.
Remote	This option is available only when the **Use Remote Status** checkbox has been selected. If selected, this option forces the calling OPC client's **GetStatus** call to the end OPC server. The status of the end OPC server will be returned. If there is a line interruption or some other form of communication failure between the two halves of Tunneller, the calling OPC client may hang and become unresponsive (it is a "blocking" call). Whether the OPC client hangs or not is entirely dependent on the OPC client and its implementation. If a communication failure is detected, Tunneller's status will be set to failed. In subsequent **GetStatus** calls, the status field of the status structure returned by Tunneller will be set to failed.

Table 3 - GetStatus Configuration Options

Note: Starting in version 3.1.0.0, when the completion of a request sent by the SSC takes too long, the SSC periodically sends "Keep-Alive" messages. The default time interval value is equal to half of the **timeout** parameter configured on the *tunneller.ini* file on the SSC. If both the CSC and SSC are of version 3.2.0.0 and higher, the time interval is equal to half of the **Communication Timeout** parameter which is configured using the Client-Side Gateway Configuration Utility.

For example, adding 15,000 items can take the OPC server 15 seconds to perform. In this scenario, the SSC will send Keep-Alive messages every 1.5 seconds (if the **Timeout Parameter** equals the default value of ***3*** seconds). If the CSC does not receive any message during the time interval set by the **Communication Timeout** parameter (which also has a default value of ***3*** seconds), the CSC detects communication failure, disconnects from the SSC and tries to reconnect during the time defined in the **ReconnectTime** parameter (in *TunnellerOpts.ini* file). If reconnection is not successful, then the add items request fails and the failed result is returned to the OPC client.

Encryption, Compression, User Impersonation, and Restriction of the Access to OPC Servers

Encryption

It may be in the interest of the user to encrypt data on the Server-Side machine by selecting which computers may connect to the end OPC servers through Tunneller. Furthermore, this communication may be further regulated by the use of encryption keys. Since the Server-Side machine contains the data that the Client-Side machine wishes to access, the option to operate in a more secure mode is made available on the Server-Side.

There are two security modes in which Tunneller may operate:

- **Open** - no encryption will be used and there is no restriction as to which computers may connect through Tunneller. By default, Tunneller will operate in ***Open*** mode.
- **Encrypted** - encryption may be used and the user will be able to control which computers are permitted to connect through Tunneller.

The information contained in this section is useful for those users who wish to operate in ***Encrypted*** mode.

Encryption is controlled by the **Client-Side Gateway Key Manager** (which is part of the CSC), and the Server-Side Gateway configuration tool (which is part of the SSC). Both of them are installed by default in ***C:\Program Files\Matrikon\OPC\Tunneller*** in the ***`Client-Side Gateway`*** and ***`Server-Side Gateway`*** subfolders, respectively.

To access the **Server-Side Gateway Configuration Tool** on the Server-Side computer, click on **Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller** -> **Server-Side Gateway Configuration Tool**.

To access the **Client-Side Gateway Key Manager** on the Client-Side computer, click on **Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller** -> **Client-Side Gateway Key Manager**. Alternatively, the **Client-Side Gateway Key Manager** is accessible using the Client Configuration Tool (**Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller** -> **Client-Side Gateway Config**).

Once the CCT is open (see Figure 4), the **Client-Side Gateway Key Manager** may be opened by pressing the second button on the toolbar (the key icon ), or by pressing **Ctrl+K**, or by selecting the **Open Key Manager** option from the **File** menu.

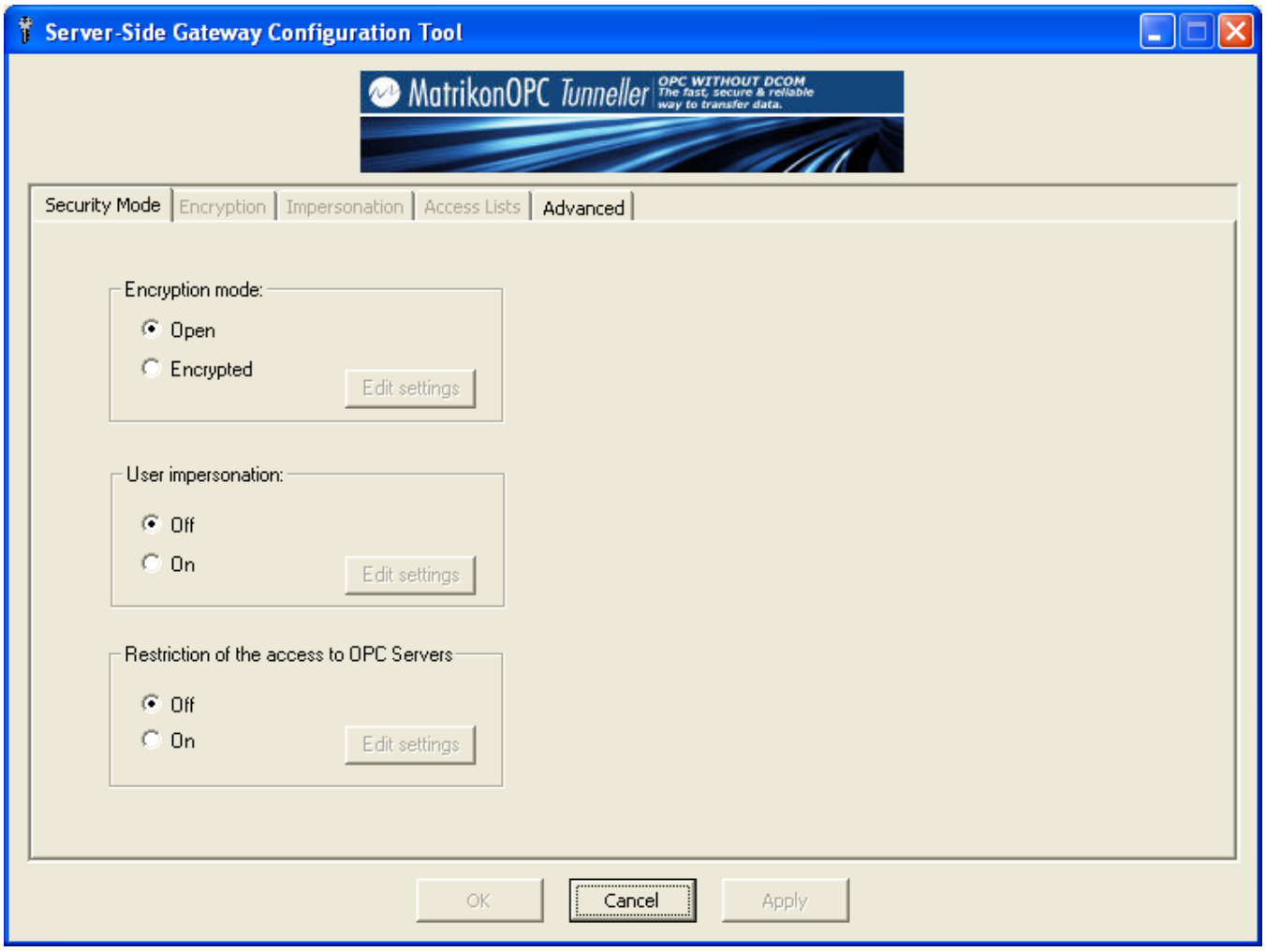


Figure 11 - Server-Side Gateway Configuration Tool (Security Mode Tab page)

Configuring Encryption Settings Using Server-Side Gateway Configuration Tool

The **Server-Side Gateway Configuration Tool** allows the user to set the Encryption mode to either ***Open*** (the default mode) or ***Encrypted*** mode. When ***Open*** mode is selected (Figure 11), the **Encryption** tab is disabled and Tunneller communication will not be encrypted.

When ***Encrypted*** mode is selected, the **Encryption** tab is enabled (Figure 12), allowing the user to configure key mappings which will regulate Tunneller encryption. Encryption fields are described in Table 4.

On the Tunneller SSC, the encryption key is read when a new communication session is created. Therefore, modifications to key mappings on the SSC will not affect existing connections. To apply modifications on the SSC, connected sessions should be recreated. Recreation of a session can be achieved from the Tunneller CSC. If all OPC clients connected to the particular end OPC server disconnect from the CSC, this will cause the disconnection of the CSC from the SSC. When the OPC client or clients connect again, the modifications will take effect. Alternatively, if a situation occurs such as no access to the OPC client machine, then restarting the Tunneller SSC service will disconnect clients and apply the SS key mappings on start up. An existing connection will continue to communicate using their old encryption settings until the session has ended.

Note: For successful communication to occur, both the SSC and the CSC must be set to the same mode (i.e., set both to either ***Encrypted*** or ***Open***). If the SSC and CSC are set to ***Encrypted*** mode, then the encryption key must match on both ends.

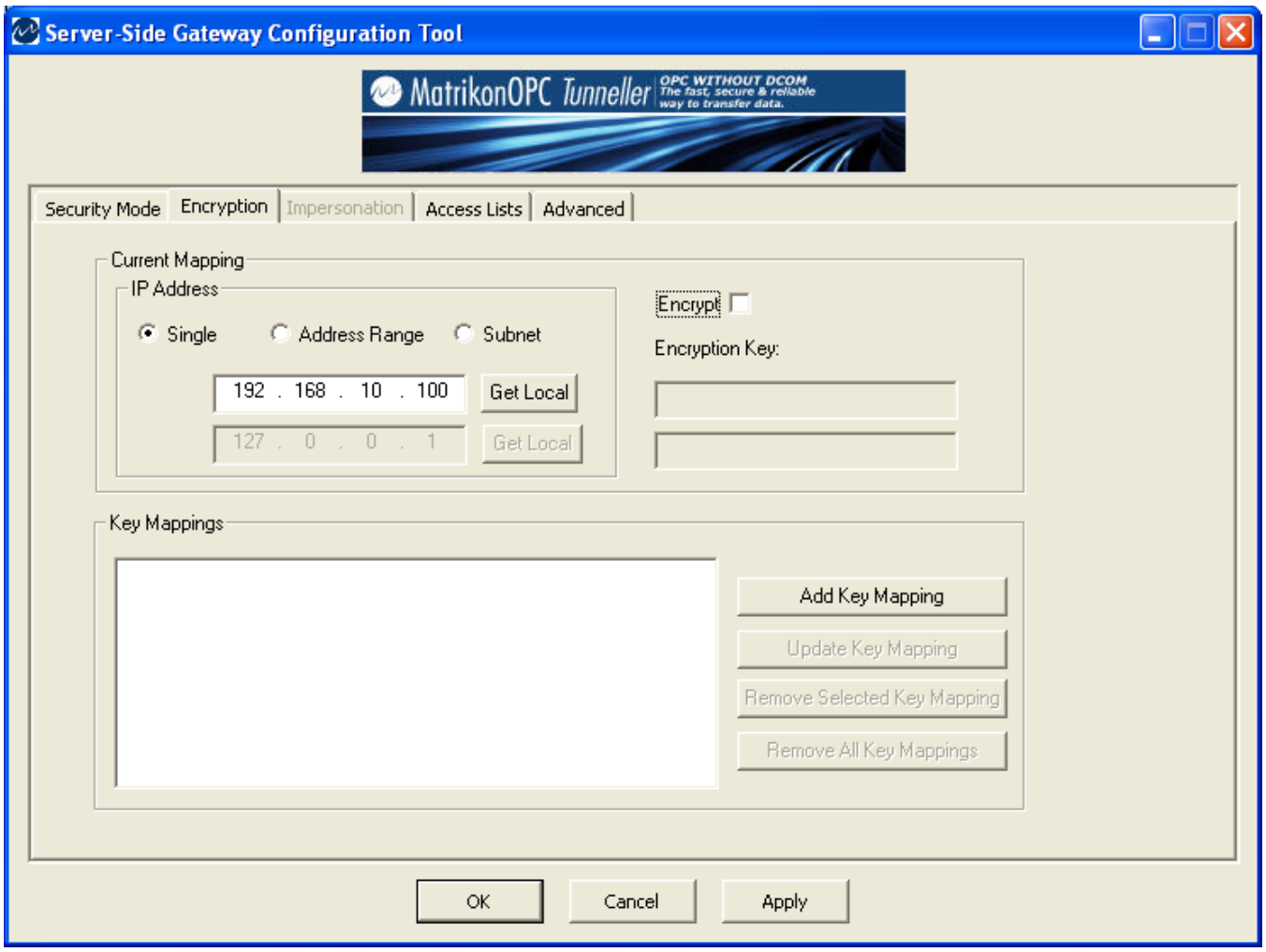


Figure 12 - Server-Side Security Configuration Tool (Encryption Tab)

Option	Description
IP Address	To allow for Tunneller communication between the Tunneller SS machine (the machine where the **Server-Side Gateway Configuration Tool** is located, with IP address *x.x.x.x*) and the Tunneller CS machine (with IP address *y.y.y.y*), the CS machine IP address *y.y.y.y* should be entered in the IP address field and an encrypted key must be created. Similarly, in the CS computer's **Client-Side Gateway Key Manager**, there should be a key mapping containing *x.x.x.x*, and the encryption key must be the same for communication to be successful. Address can be entered in three different ways: • *Single* – IP address is defined exactly. • *Address range* – start and end addresses of the range are defined. In this case the key will be used for all IP addresses belonging to the range.

Option	Description
	• *Subnet* – the mask is entered in the first field and the subnet address is entered in the second field. In this case if the result of a bitwise AND operation of the CSC host's address and *Mask* is equal to the result of bitwise AND operation of *Address* value and *Mask*, then the corresponding key will be used for that particular address. For example, the Mask = 255.255.0.0 and the Address = 192.168.0.0. In this case for all hosts belonging to the 192.168.0.0 local subnet the same key will be used. Using arrows the rightmost bit of *Mask* can be shifted to the left or to the right. The first click on **Get local** button changes *Address* filed to local IP address. Second click changes *Address* field to the result of bitwise AND operation of local IP address and *Mask*. Note that for a subnet mask to be valid, its leftmost bits must be set to '1'. Conversely, the rightmost bits in a valid subnet mask must be set to '0', not '1'. So all valid subnet masks contain two parts: the left side with all mask bits set to '1' (the extended network portion) and the right side with all bits set to '0' (the host portion). Note that *Subnet* type entries are sorted in certain order taking into account how many bits define extended network portion, for example: ○ 255.255.0.0 168.192.0.0 ○ 255.0.0.0 168.0.0.0 ○ 255.255.0.0 192.168.0.0 ○ 255.0.0.0 192.0.0.0
Get Local	If this button is pressed, the **IP Address** field will display the local computer's IP address.
Encryption mode (only in Server-Side Gateway Configuration Tool, Security Mode tab)	The default setting is ***Open***. When this mode is selected, the **Encryption** tab is disabled. When the ***Encryption*** option is selected, the user can select which computers may connect through Tunneller and may set up encryption.
Encrypt	When setting up a key mapping, it must be decided if Tunneller communication for the selected IP address will be encrypted. If encryption is desired, check this box; otherwise, ensure it is unchecked.
Encryption Key	When setting up a key mapping, if encryption will be used for Tunneller communication with that specific IP address or range of addresses, an encryption key may be entered. For Tunneller communication to be successful, the same key must be entered on both the **Client-Side Gateway Key Manager** and the **Server-Side Gateway Configuration Tool** for that particular key mapping. An Encryption key may contain letters, numbers, and special characters available on a keyboard.

Option	Description
	The Encryption key must be entered twice for validation. If keys are different, **Update/Add Key Mapping** buttons will be disabled. If the **Encryption key** field is empty, then default hard-coded key will be used.
Key Mappings	This field will display the user-created key mappings. Key mappings control which computers may connect through Tunneller and contain the encryption keys being employed. The format of a key mapping is either: • *IP address*, *<#bits encryption>,* • or *IP address <Non encrypted>* , which indicates that no encryption will be used for the listed IP Address. Here *IP address* can be just single address or range of IP addresses or the subnet mask and address combination. Clicking on a key mapping will fill the current mapping fields with the selected key mapping settings. Note that the list of keys is ordered in the following order: 1. Single address in ascending order. 2. Address ranges in ascending order of "From" address. 3. Subnets in ascending order of address value. The first found entry in Key mapping is used for the given IP address.
Add Key Mapping	Once the user has filled the IP address information and encryption key (optional), pressing this button will create a new key mapping. If a key mapping containing the specified IP addresses already exists, this button will not be enabled.
Update Key Mapping	This button is used to update the selected key mapping with the values from fields **Encrypt**, **Encryption key** and **IP address**. If the values on selected key mapping have not changed, this button will be disabled.
Remove Selected Key Mapping	To remove a Key mapping, select it from the list of Key mappings and press this button.
Remove All Key Mappings	Pressing this button will remove all Key mappings from the list.
OK	This button acts exactly in the same way as the **Apply** button (changes are saved). The difference is that after saving of key mappings the dialog window will be closed.
Cancel/Close	If key mappings were modified but not saved, then this button is labelled ***Cancel***. Otherwise, its label states ***Close***.
Apply	This button is enabled if changes to key mappings have been made. **Note:** If key mappings were modified, but not saved, and later their previous values are recovered, then the **Apply** button will be disabled.

Table 4 - Encryption Options

The encryption level depends on the length of the user-entered encryption key. Encryption keys will be padded out to the appropriate length (the greatest number of characters in that range) automatically. For example, a 12-character encryption key will be padded out to 16 characters.

Number of Characters in Encryption Key	Number of Bits Encryption
1 – 16	64
17 – 24	96
25 – 32	128

Table 5 - Encryption Key Length vs. Number of Bits

Client-Side Gateway Key Manager

The **Client-Side Gateway Key Manager** (see Figure 13) allows the user to create key mappings on the CSC which will correspond to SS key mappings, in the event that the **Server-Side Gateway Configuration Tool** is configured for ***Encrypted*** mode. The fields in **Client-Side Gateway Key Manager** are the same as encryption related fields from Encryption mappings tab page in the **Server-Side Gateway Configuration Tool** described in Table 4.

In ***Encrypted*** mode, both the **Client-Side Gateway Key Manager** and the **Server-Side Gateway Configuration Tool** must be configured properly with matching encryption keys.

Modifications of encryption key mappings become effective immediately on CSC.

Example

A user wishes to use Tunneller to allow OPC Clients on Computer 1 to obtain data from end OPC Servers on Computer 2. Computer 1 with IP address 192.168.10.100 is the Client-Side, and Computer 2 with IP address 192.168.10.200 is the Server-Side. The Client-Side Gateway on Computer 1 must contain Computer 2's IP address in a key mapping. Similarly, the **Server-Side Gateway Configuration Tool** on Computer 2 must contain Computer 1's IP address in a key mapping. Furthermore, the same encryption settings must be used. If the key mapping on the CSC is: *192.168.10.200, <64 bit encryption>* (see Figure 13), then the corresponding key mapping on the SSC must be: *192.168.10.100, <64 bit encryption>* (see Figure 14), using the same encryption key.

Note: In the example for the **Server-Side Gateway Configuration Tool** (see Figure 14) all of the lines under Key mappings are used to represent the CSC IP address. In the first line the IP address is defined exactly. In the second line the range of addresses from 192.168.10.0 to 192.168.10.255 is defined. In the third line, the subnet mask and address are defined.
Bitwise AND of CSC IP address and *Mask* is:

192.168.10.100 & 255.255.255.0 = 192.168.10.0

Bitwise AND of *Mask* and *Address* is:

255.255.255.0 & 192.168.10.100 = 192.168.10.0

Results of both operations are equal, so this line could be used to represent given CSC address too.

In such a situation, the first found key is used (line with 64-bit encryption).

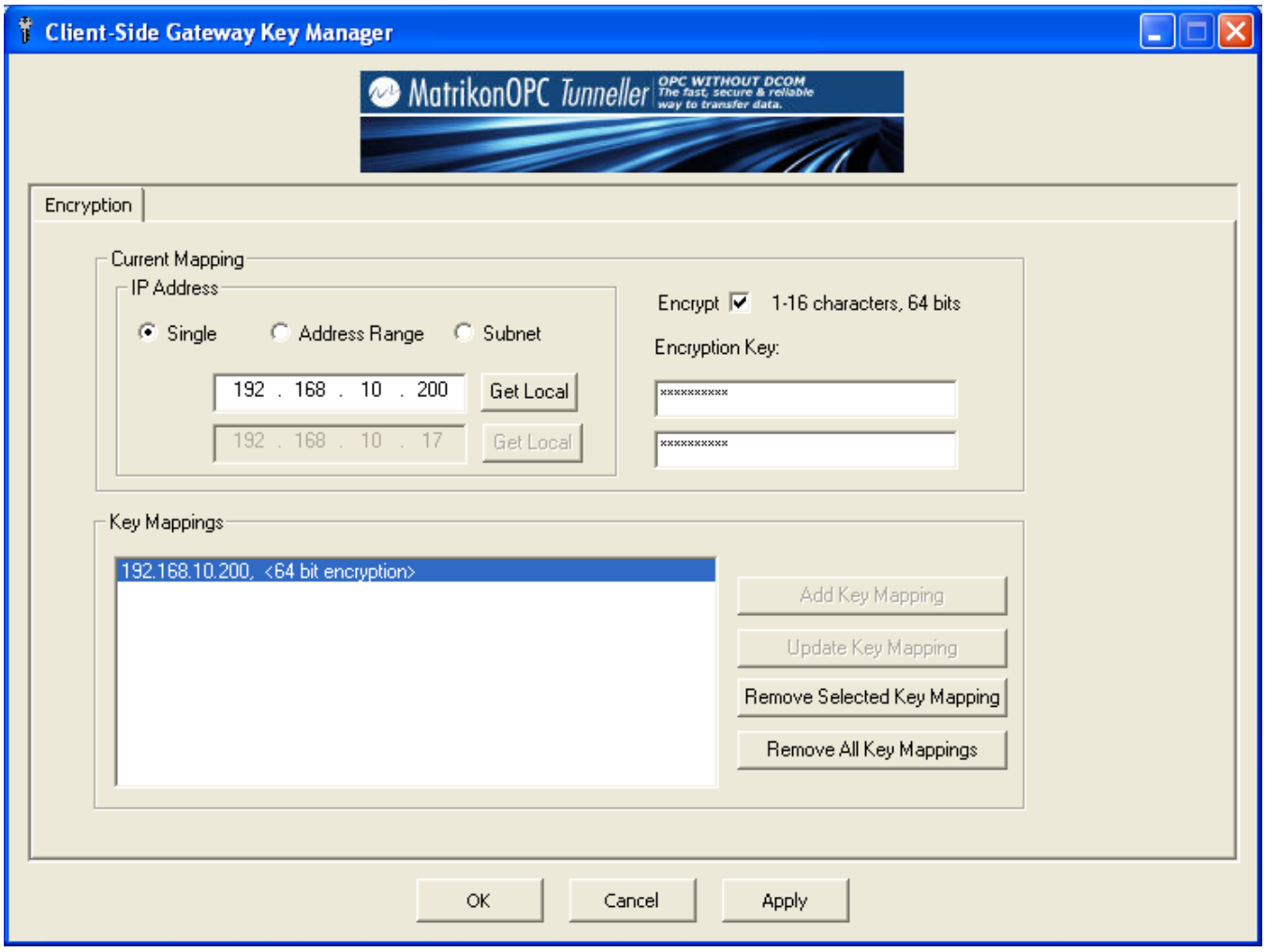


Figure 13 - Example: Client–Side Gateway Key Manager

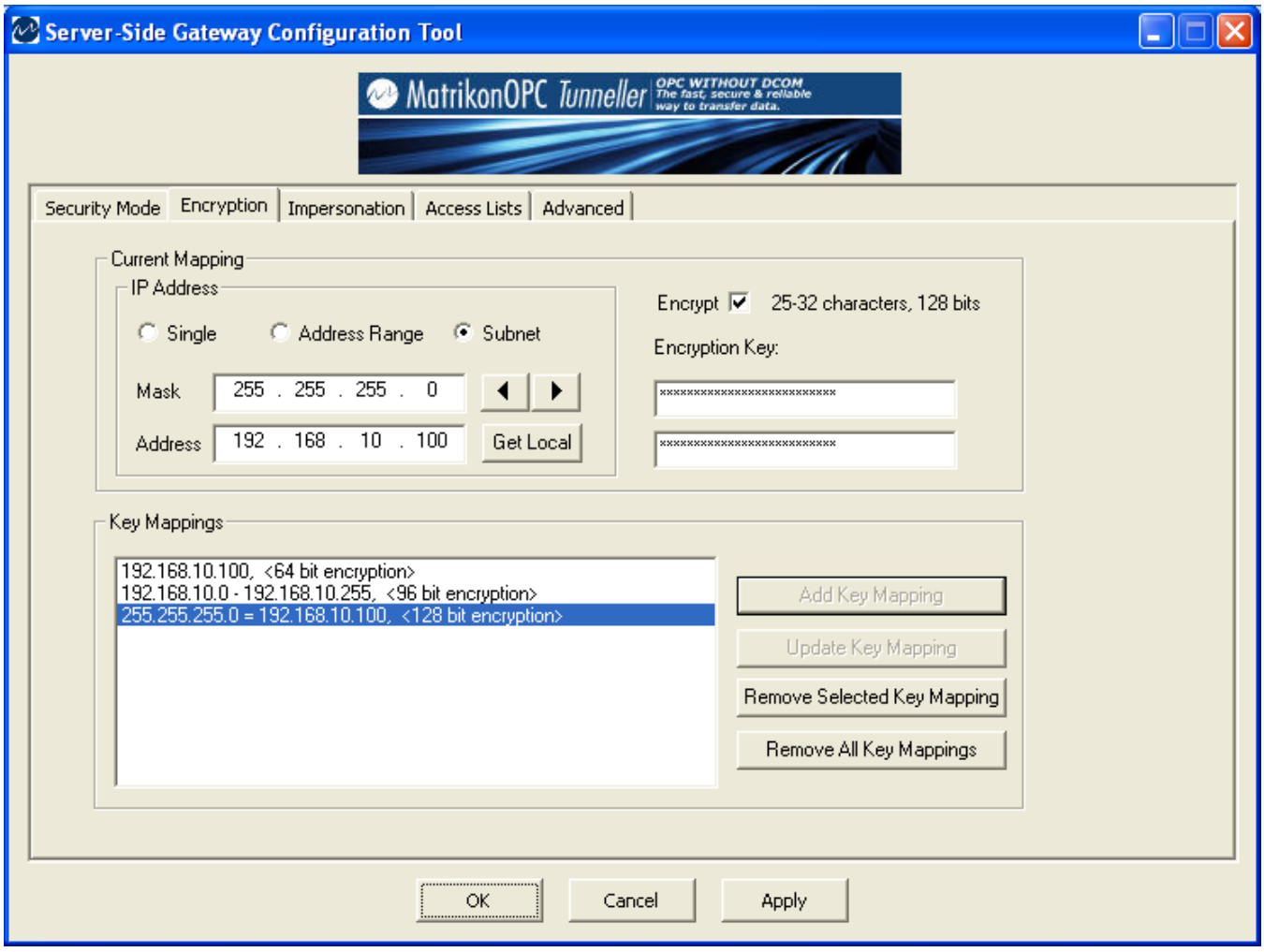


Figure 14 - Example: Server-Side Gateway Configuration Tool (Encryption Tab)

Compression

Compressing messages may speed up data transfer considerably if the communication channel has low bandwidth. Compression was introduced in version 3.0.0.0 of Tunneller. If compression is enabled in an attempt to communicate with an earlier version of the Tunneller SSC then this option is ignored and does not affect communication behaviour.

The compression option is configured in the **Client Configuration Tool** by selecting the **Use Compression** check box (Figure 15).

The **Use Compression** option is stored in the Windows system registry and is read each time an OPC client connects to the CSC.

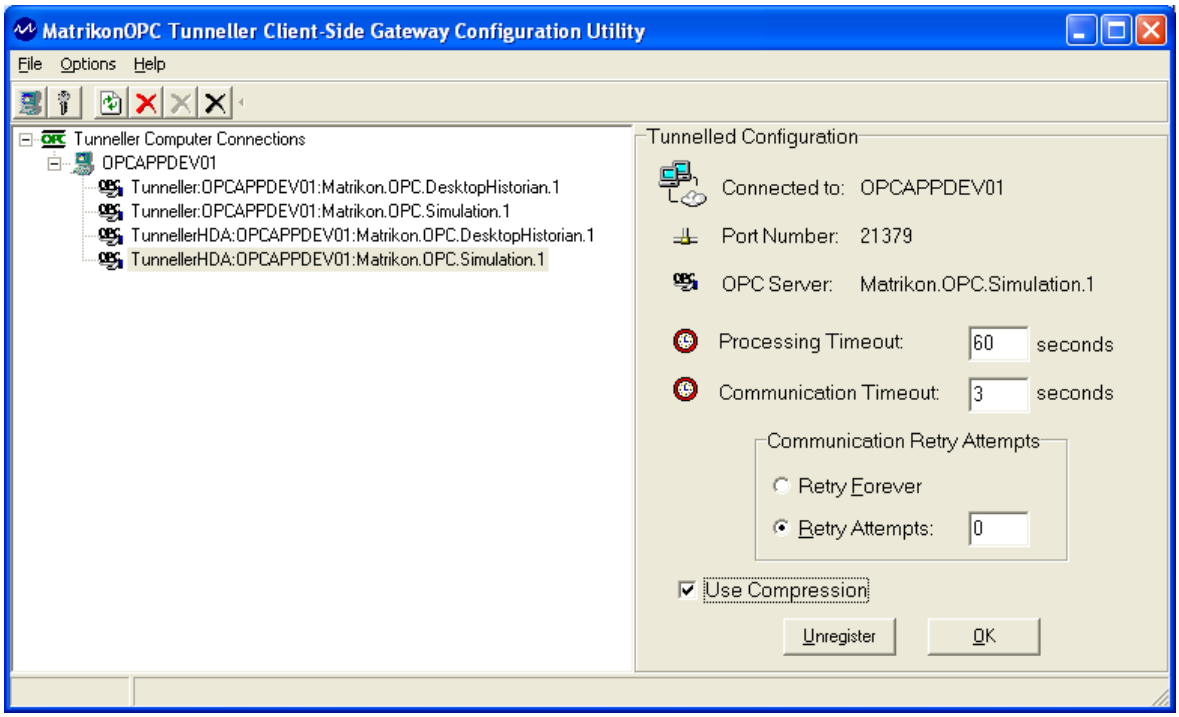


Figure 15 - Client-Side Configuration Utility (with Use Compression Checkbox)

User Impersonation

Starting in version 3.1.0.0, MatrikonOPC Tunneller has user impersonation functionality. By default this feature is turned off. If turned on, Tunneller CSC retrieves information about domain and user name under which OPC client connects, and passes that information to the Tunneller SSC. On the server side, Remote user to Local user mappings can be configured. Tunneller SSC looks for the entry for Remote user on User mappings. If the entry is found, it takes Local user for that entry and performs impersonation using its domain, user name and password for the thread communicating with the end OPC server. As a result, the end OPC servers can behave differently based on what user is connected, for example restrict access.

User impersonation settings are configured using **Server-Side Gateway Configuration Tool**.

The impersonation feature can be turned ***ON*** or ***OFF*** using the **Security mode** tab of the **Server-Side Gateway Configuration Tool** (Figure 11). If impersonation is turned ***ON***, the fields on **Impersonation** tab page (Figure 16) become enabled.

Note: Impersonation can also be turned on or off through the *tunneller.ini* file, ***UseImpersonation*** parameter under the **TCConnection** section. Its default value is ***0*** (i.e., user impersonation is ***OFF***). User impersonation mappings can be configured only using the **Server-Side Gateway Configuration Tool**.

The behaviour of the system in case no entry is found for the provided Remote user or the Remote user is not defined (this can be possible if older version of Tunneller CSC is connected) depends on the state of **Use Default account** checkbox. If it is turned ***OFF***, then the connection will be rejected. If it is turned ON, the entry for **[Default]** Remote user will be used. If at the time when the checkbox is turned ***ON*** and no such entry is found, the **Edit User Mapping** window is opened and the entry will be added.

Local user corresponding to **[Default]** Remote user can be configured either as a specific local user account or as **[Default]**. If it is configured as **[Default],** then the user account under which

Tunneller SSC is running will be used for impersonation. If **User Impersonation** is turned ON but there are no mappings configured, and the **Default Account** option is turned OFF, a warning message pops up when changes are applied.

To add a new entry into the User mappings:

1. From the **Server-Side Gateway Configuration Tool** window, select the **Impersonation** tab, and click on the **Add** button.
2. The new **Edit User mapping** window (Figure 17) is displayed.
3. Enter the **Remote user**, **Local user**, and **Password**.

   Note: Remote user should be entered as ***Domain\User name*** for domain accounts or ***Computer name\User name*** for machine specific accounts. You can also specify the hostname of the end OPC client's machine using the ***Domain\User name:hostname*** syntax, where ***hostname*** can be the fully qualified domain name (FQDN) hostname or simply the machine name. The **Local user** field requires the user name only.
4. Select the **OK** button.

   Note: All impersonation mappings including passwords are stored in the configuration file using encryption.

The currently selected entry can be edited by double-clicking your mouse on the entry or by selecting the **Edit** button. To delete an entry or entries from the User mappings, select one or more entries and click on the **Delete** button.

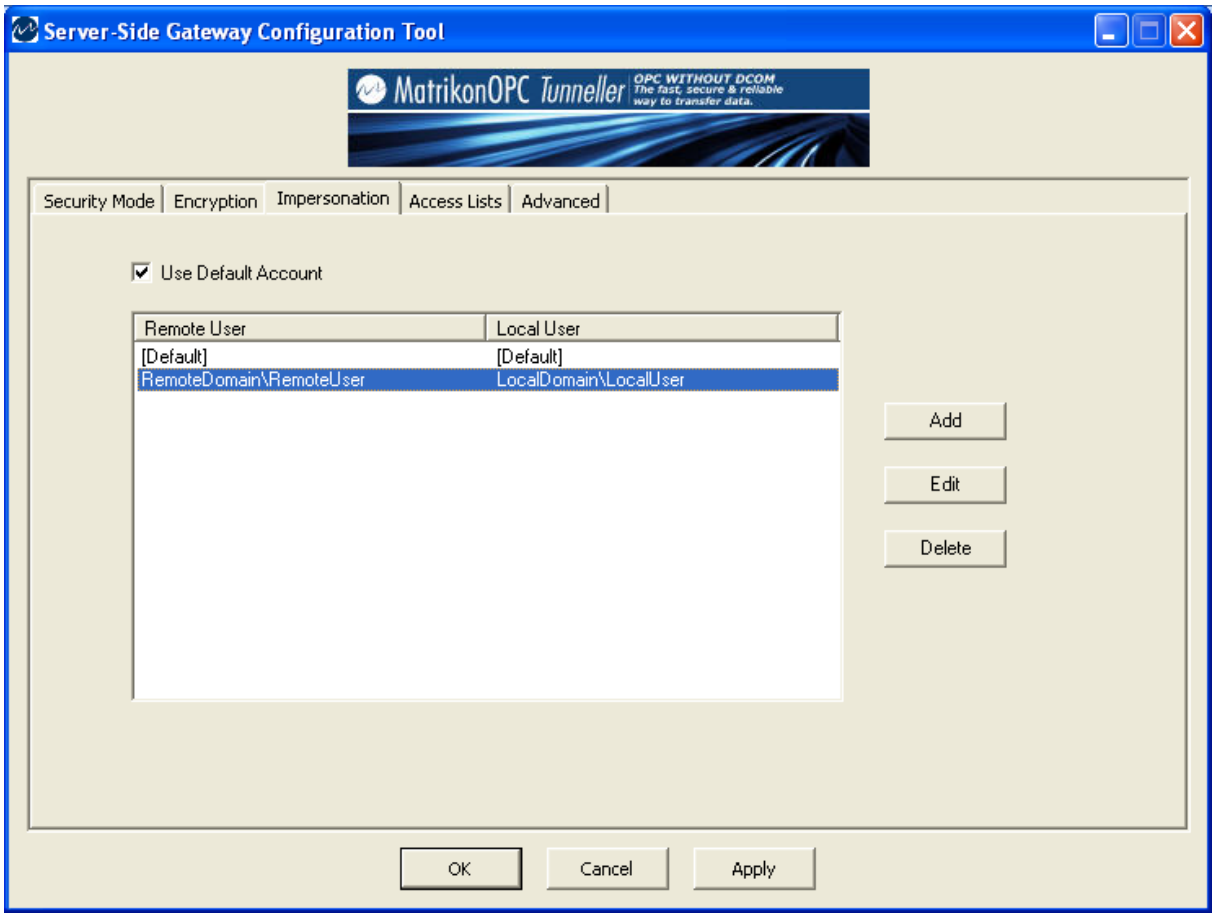


Figure 16 - Server-Side Gateway Configuration Tool (Impersonation Tab)

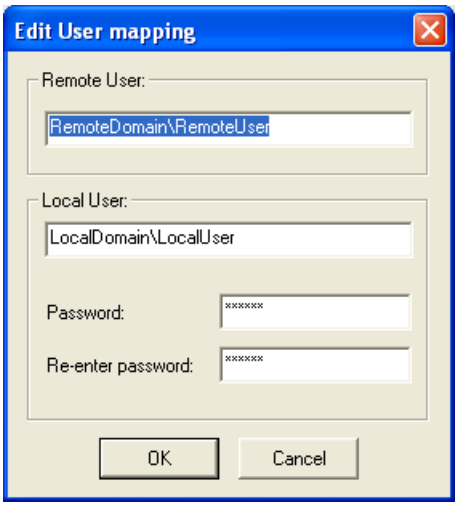


Figure 17 - Edit User Mapping Window

OPC Server Access Restriction and MatrikonOPC Security Gateway Integration

Starting in version 3.1.0.0, Tunneller provides functionality to restrict access to OPC servers installed on the Server-Side Component depending on Remote client. Remote clients can be authenticated either by their local IP Address (on their local network), or a fully-qualified host name, or Domain\User name, or a combination of these three fields. Each Remote client can have a configured list of accessible OPC servers. If the Remote client is unknown (i.e., there is no configuration for it), then the default list can be configured (all authentication fields set to [Default] on Remote clients list for this entry).

Restriction of the access to OPC servers can be turned on from the **Security Mode** tab of the **Server-Side Gateway Configuration Tool** (Figure 11).

Settings for what fields are used to authenticate the Remote client and what Remote clients can access which OPC servers, are configured using the **Access Lists** tab (Figure 18). Components of this tab are described in Table 6.

Starting in version 3.1.2.0, OPC server access restriction is implemented differently in cases where the **MatrikonOPC Security Gateway** is installed on the same box as the Tunneller Server-Side Component. In that situation, Tunneller SSC returns only the ProgID of MatrikonOPC Security Gateway as the list of installed OPC servers. Therefore, remote OPC clients can only connect to OPC servers through MatrikonOPC Security Gateway. That provides functionality to control access rights per remote user at the OPC Items level.

If the **Server-Side Gateway Configuration Tool** detects that the MatrikonOPC Security Gateway is installed and has a valid license (including a demo license), then the **Access Lists** tab becomes invisible. As well, on the **Security Mode** tab the text **Controlled by Security Gateway** appears under the **Restriction of the Access to OPC Servers** group.

Note: When the Server-Side Gateway Configuration Tool starts up, a check is performed to see if the MatrikonOPC Security Gateway is installed and licensed. Therefore, if licensing conditions are changed, the Server-Side Gateway Configuration Tool should be restarted for changes to take effect.

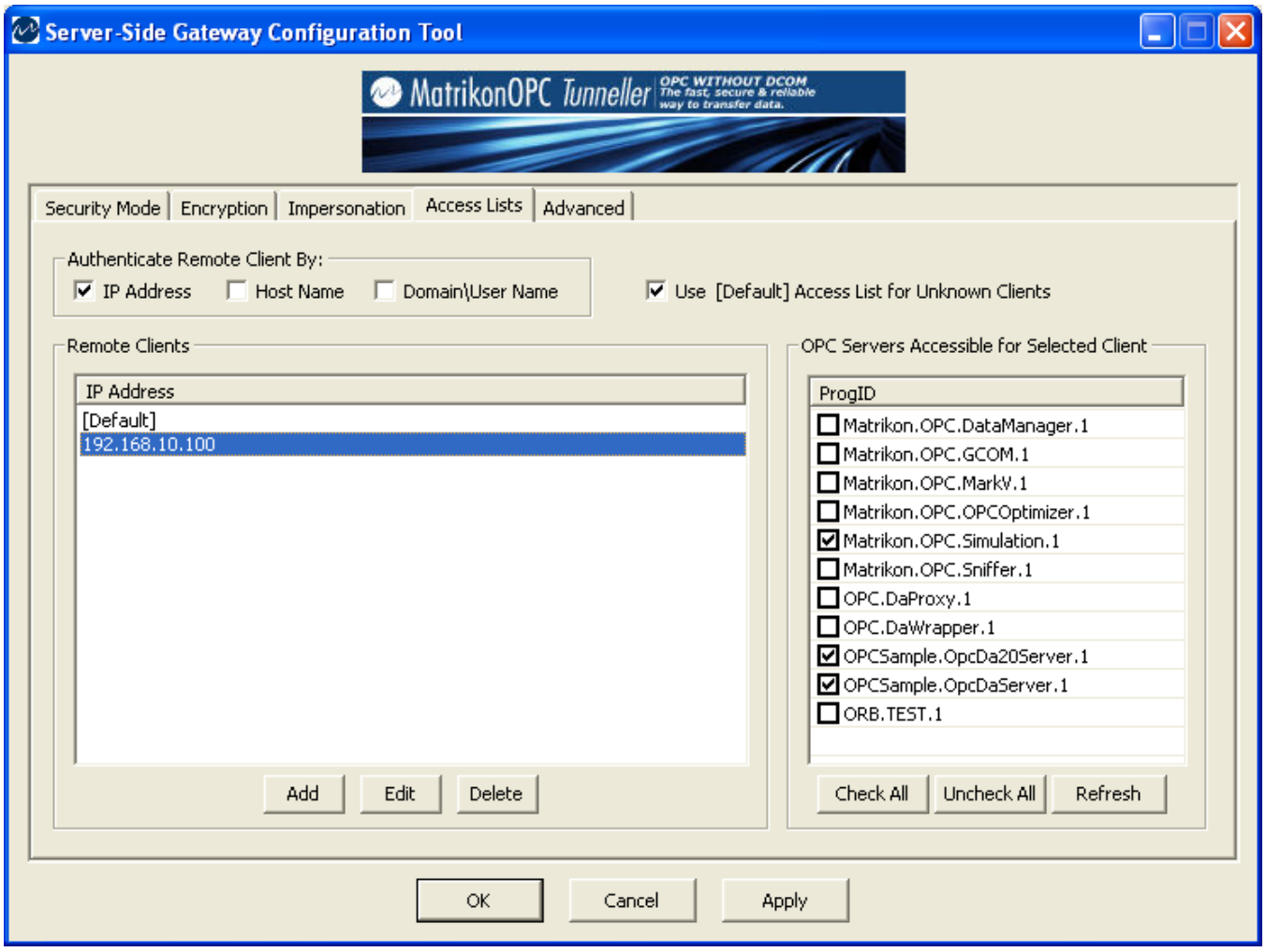


Figure 18 - Server-Side Gateway Configuration Tool (Access Lists Tab)

Component	Description
Authenticate Remote Client By	Checkboxes defining what data is to be used to authenticate remote client: **IP address**, **Host name**, or/and **Domain\User name**. At least one checkbox must be checked.
Remote Clients	The list of entries authenticating Remote clients. The number of visible columns depends on the selected **Authenticate Remote client by** checkboxes. **Note:** For each combination of selected **Authenticate Remote client by** checkboxes, separate lists are stored on the **Remote clients** list.
Use [Default] Access List for Unknown Clients	This checkbox defines the behaviour of the system in case the Remote client could not be authenticated (unknown). If selected, then the list of accessible **OPC Servers for [Default]** entry will be used (all fields have [Default] value for this entry on Remote Clients list). Otherwise, access is refused and the empty list is returned to the **Client-Side Gateway Configuration Tool**.
Add	Adds a new entry into the list of Remote clients. Opens a new window.
Edit	Opens the currently selected entry from the list of Remote clients for

Component	Description
	modifications.
Delete	Deletes the currently selected entry from the list of Remote clients.
OPC Servers Accessible for Selected Client	The list of OPC servers installed on local computer (where Tunneller SSC is hosted). Select the applicable checkboxes to determine whether the corresponding OPC server is accessible to the currently selected Remote client or not. OPC servers are identified by their ProgID.
Check All	Marks all OPC servers in the list.
Uncheck All	Un-marks all OPC servers in the list.
Refresh	Refreshes the list of OPC servers, using OPCEnum service or direct access to the system registry (depending on **Browse Registry** parameter defined on *tunneller.ini* file).

Table 6 - Control of the Access to OPC Servers Window Components

New entries to the Remote clients list are added by clicking on the **Add** button. Changing the selected entry in the list of Remote clients can be done by selecting the **Edit** button. In either situation, the **Edit Remote client host info** window (Figure 19) is displayed. Components of that window are described on Table 7.

Notes:

- The [Default] entry on **Remote clients** list cannot be edited or deleted.
- All fields selected for use for authentication, must have non-empty values. If the fields used for Remote client authentication are modified (for example, initially only the IP address was used, but later IP address and Domain\User name are used), then previous settings are still stored in the configuration file, but they will not be used or displayed on GUI. Only entries which have non-empty IP Address and Domain\User name and empty host name will be used and will be visible on the **Remote clients** list.
- If **Access Restriction** is turned ON, but there are no accessible OPC servers, a warning message pops up when changes are applied.

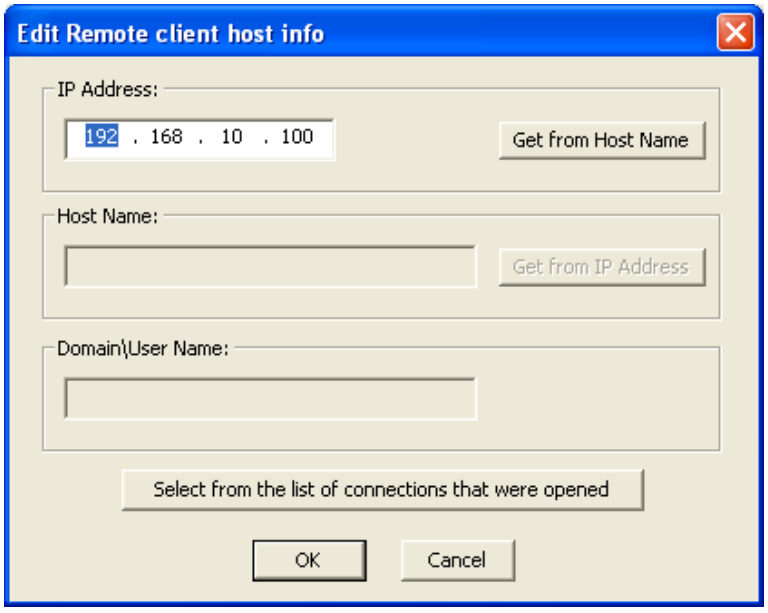


Figure 19 - Edit Remote Client Host Info Window

Component	Description
IP Address	IP address of Remote client (on its local network). Note that if there is a router between CSC and SSC hosts, one SSC log file client's IP address can be shown differently). Option is disabled if the **IP address** checkbox is not selected in the **Authenticate Remote client by** group of the **Access Lists** tab.
Get from Host Name	Get IP address by resolving **Host name**. If the **Host name** option is disabled, click on this button to enable it. A value can then be entered into the **Host name** field. A second click to this button will retrieve the **IP Address**. **Note:** This can be time consuming. While processing, the button will be disabled and its text will be changed to "*Getting…*".
Host Name	Fully qualified Host name of Remote client. Option is disabled if the **Host name** checkbox is not selected in the **Authenticate Remote client by** group of the **Access Lists** tab.
Get from IP Address	Get IP address by **IP address**. If the **IP address** option is disabled, click on this button to enable it. A value can then be entered into the **IP address** field. A second click to this button will retrieve the **Host name**. **Note:** This can be time-consuming. While processing, the button will be disabled and its text will be changed to "*Getting…*".
Domain\User Name	Domain\User name of the windows user account under which the OPC client application is running. **Note**: User account, under which the OPC client application runs, might be different than the currently logged-on-to remote host user. For example, if OPC Client Applications was launched using **Run As** utility or if it runs as a service.
Select from the list of connections that were opened	Click on this button to open the **Select Remote Client info from the list of connections** window (Figure 20). Tunneller SSC keeps track of which Remote clients were connected to it, including connections from the **Client-Side Gateway Configuration Tool**. The window opened by this button contains a list of Remote client authentication data that can be selected and used to fill corresponding fields. Entering Remote client host information by selecting from the list of connections ensures that the authentication fields have correct values and the configuration process is sped up. **Notes:** • If no connections have yet been accepted by SSC, the connections list is empty and the button is disabled. • If an older version of Tunneller CSC was connected to SSC, the list of connections will have a line with blank fields as older versions do not pass Remote client information to the SSC.
OK	Saves changes in memory and closes the window. **Note:** Changes will take effect when either the **OK** or **Apply** button on the main window is selected.

Component	Description
Cancel	Closes the window without saving changes.

Table 7 - Edit Remote Client Host Info Window Components

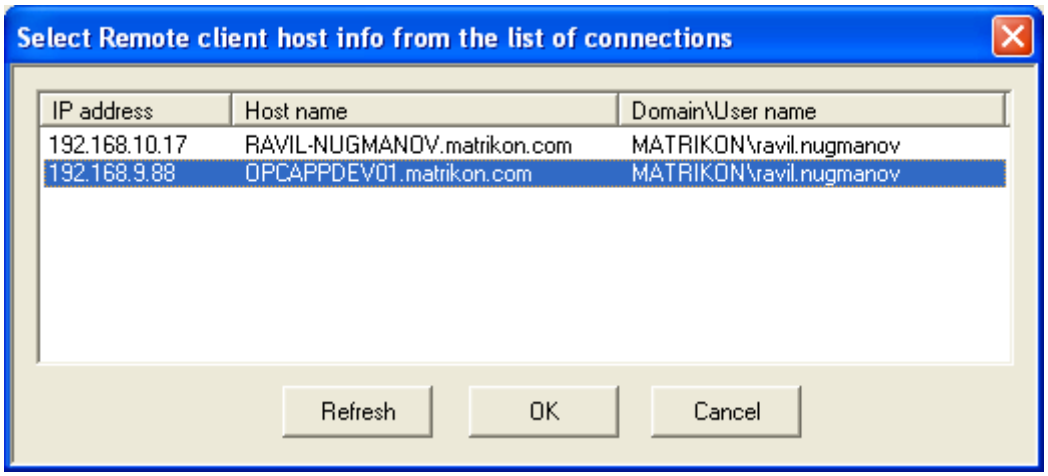


Figure 20 - Select Remote Client Host Info From The List Of Connections Window

To ensure that the Remote client host information is entered correctly, it is recommended that you use the **Select Remote client host info from the list of connections** window.

Advanced SSC Settings

Advanced configuration settings for the Server-Side Component can be modified using the **Advanced** tab on the **Server-Side Gateway Configuration Tool** window (Figure 21). Components of this tab are described in Table 6.

Advanced settings are stored in the *tunneller.ini* file.

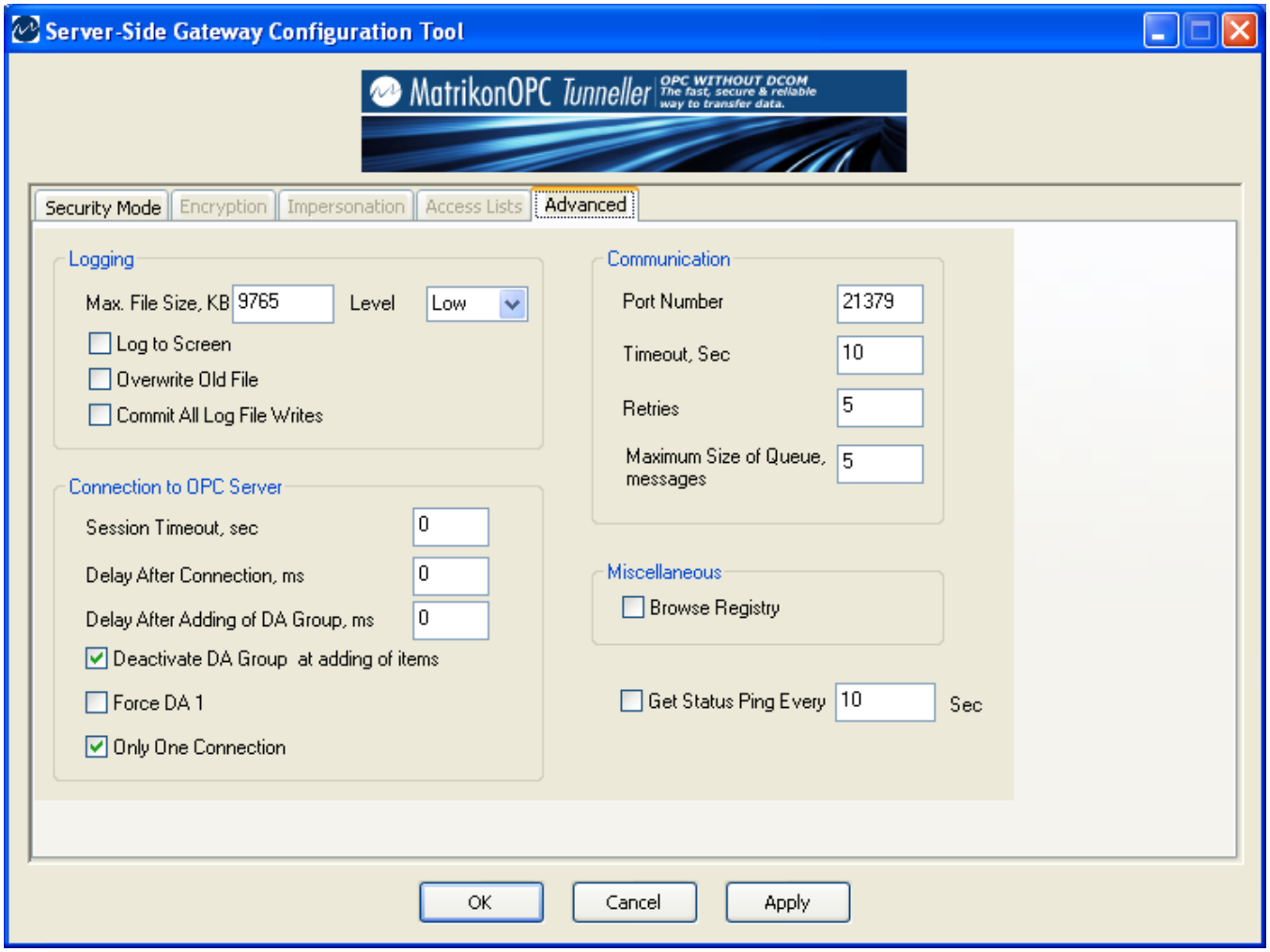


Figure 21 - Server-Side Gateway Configuration Tool (Advanced Tab)

Component	Description
Max File Size, KB	Maximum size of the *tunneller.log* file in Kbytes.
Level	Log level.
Log to Screen	If this checkbox is selected, Tunneller SSC logs messages to the screen. It does not disable logging to file.
Overwrite Old File	If this checkbox is selected, the previous log file's copy is not made. Otherwise, backup is made named as *tunneller.log.bak*.
Commit All Log File Writes	If this checkbox is selected, all information written to the log file is immediately written to disk instead of being cached. **Note:** This can slow down the operation of Tunneller as operations must

Component	Description
	wait for log lines to be flushed to disk before continuing.
Port Number	The TCP port address to which SSC listens for incoming connections. **Note:** To successfully communicate, the port number configured on Server-Side Component must match with the port number configured on Client-Side Component. If this option is changed, the Tunneller service must be restarted.
Timeout	Communication timeout in seconds. If the CSC version is 3.2.0.0 or higher, this value is ignored and the CSC's **Communication Timeout** option is used.
Retries	Number of retries after communication failure. If the CSC version is 3.2.0.0 or higher, this value is ignored and the CSC's **Retry Attempts** option is used.
Maximum Size of Queue	Maximum number of outgoing messages on the communication messages queue. See **Update of large amount of items fails** in the **Troubleshooting** section for more details. The default value is ***5***.
Session Timeout	Used when communication failures occur. During the **Session Timeout** period, the connection to the end OPC server is not closed and the session stays open (i.e., OPC groups and items are not deleted). When the Client-Side Component reconnects to the Server-Side Component, that session can be reused so that recovering does not take a long time. This option should have the same value as the **ReconnectTime** option configured on Client-Side Component versions prior to 3.2.0.0. If both CSC and SSC are of version 3.2.0.0 or higher, then **ReconnectTime** defined on CSC is used as a **Session Timeout**.
Delay After Connection	Defines the delay (in milliseconds) after connection to the end OPC server.
Delay After Adding of DA Group	Defines the delay (in milliseconds) after adding of group to OPC DA server.
Deactivate DA Group at adding of items	Deactivates the DA Group before the adding of items, and activates after adding.
Force DA 1	Forces the use of DA 1 interfaces on connection to OPC DA server.
Only One Connection	If this checkbox is selected, it specifies that only one connection at a time is created to the OPC server. Usually this checkbox should be selected. It takes effect in two cases. 1. If a communication error occurs, then the session is waiting for reconnection for a defined session timeout period. But, the Client-Side Component can send a request to create a new session at that time. If this option is selected, then the Server-Side first waits for disconnection from the OPC server and deletion of the previous session, and then only creates a new session. Therefore, only one connection to the end OPC server will exist at any time (that can be required if the end OPC server can accept only one connection at a

Component	Description
	time, for example due to licensing). Otherwise, for a session timeout period, more than one open connection to the end OPC server can exist. 2. On the Windows box, only one instance of the Client-Side Component can run at the same time. But, there are installations where the Client-Side Component runs under Linux (for example, ScanTask for RTAP). In this case, multiple instances can run on the client-side and the **Only One Connection** checkbox must be cleared.
Browse Registry	If checked, Tunneller SSC accesses Windows system registry to retrieve the list of installed OPC servers. Otherwise, OpcEnum service is used. Should be turned ***ON*** for Windows NT4.
Get Status Ping	When selected, the Tunneller SSC periodically (period set in seconds) sends a Get Status request to verify whether the end OPC server is still running.

Table 8 - Advanced Tab Components

Note: Some settings on the Server-Side Component can be overridden by Client-Side Component settings if both SSC and CSC are of version 3.2.0.0 or higher, so that each session can have its own settings. Refer to Table 9 for those settings that can be overridden.

Client-Side Component Option	Server-Side Component Option
Communication Timeout	**Timeout**
Retry Attempts	**Retries**
Processing Timeout	No matching option on the SSC configuration utility GUI. Its value is used when processing a shutdown request from OPC server: SSC waits during this time for the current operation (if there is any) completion before disconnecting. If an CSC version prior to 3.2.0.0 connects to SSC version 3.2.0.0, the default value of 600 seconds is used for that session. This default value can be changed by adding the **ProcessingTimeout** option into the *tunneller.ini* file ([**TCConnection**] section).
ReconnectTime	**Session Timeout**
ConnectDelay	No matching option on the SSC configuration utility GUI. For connections from a CSC version prior to 3.2.0.0, the default value of 1000 ms is used, which can be changed by adding the **ConnectDelay** option into the *tunneller.ini* file ([**TCConnection**] section).

Table 9 - SSC Settings Overridden by CSC Settings

Connection and Reconnection

Connection Failure Scenario

Tunneller maintains a connection-oriented TCP link between the CSC and the SSC whenever there is an OPC client connected to a Tunnelled ProgID. If this link fails and is detected by the CSC or SSC, Tunneller will attempt to re-establish the link during the time period specified by the **ReconnectTime** option. The **ReconnectTime** option is configured in the *TunnellerOpts.ini* file on the Client-side Component. The default value is ***0*** indicating that no reconnection attempts will be performed.

Connection/reconnection is performed in the following order:

1. A connection is established and normal communication is underway.
2. A failure is detected (either a broken link is detected or a Communication Timeout has been passed).
3. If the SSC is still active (i.e., has not been terminated) it will save the session's current state for some configurable period of time.

   Note: For versions prior to 3.2.0.0, its value is the same for all sessions which is defined by the **Session Timeout** parameter under the TCConnection section in the *tunneller.ini* configuration file (by default, ***0*** seconds). In version 3.2.0.0, it is equal to the **ReconnectTime** option defined on the CSC (i.e., a different session can have a different session timeout depending on the CSC settings).
4. If the **ReconnectTime** option is more than ***0***, during that time the CSC tries to re-establish a connection to the SSC using a reconnect command.
   a. If the reconnect command succeeds (i.e., detects that the SSC has the correct current state available), the state information is reused and Tunneller continues on as normal. At no time during this phase has Tunneller changed the item values, qualities, or timestamps.

      Note: During communication failure period SSC handles DA OnDataChange call-backs from the end OPC server differently; only latest updates are kept in memory. When reconnection is established, SSC sends these updates to CSC as a single update. All alarms and events received during communication failure are buffered (as long as the SSC **Session Timeout** has not expired) and will be sent upon reconnection.

   b. If the reconnect command fails, Tunneller CSC waits for the delay defined by the **ReconnectDelay** parameter (default value is ***10*** seconds) and retries a reconnection during the time period defined by the **ReconnectTime** parameter (default value is ***0*** seconds).

      If reconnection could not be established during **ReconnectTime,** or communication is established but the Server-Side Component does not have a matching session in correct state, Tunneller will start full connection attempts. At this point, Tunneller will set all of its items qualities to ***bad*** informing the end OPC client that a failure has occurred.
5. The CSC will attempt to connect to SSC using the full connect command.
   a. If the connect command succeeds, the state (items, for example) maintained in the CSC will be sent to the SSC recreating the state on the SSC. Communication will

continue on as normal. Any items that had their qualities set to ***bad*** will have their qualities set to ***good*** only when a new value is received for the item.

b. If the connect command fails it will re-try the full connect command until it either succeeds or the end OPC client disconnects from the Tunnelled ProgID. Delay between retries is defined by the **ConnectDelay** option. In Tunneller SSC version 3.2.0.0, the **ConnectDelay** option also defines the delay between attempts to connect to the remote OPC server on SSC.

6. If no connection is re-established, any session state information on the SSC will be removed after the configurable time period defined by the **Session Timeout**.
7. The connection between CSC and SSC is terminated when, on CSC, there are no more connected end OPC clients during the period of time defined by the **NoClientsDisconnectionDelay** parameter in the *TunnellerOpts.ini* file. SSC will disconnect from the end OPC server immediately after the normal disconnection of CSC, or when the session has timed out (defined by the **Session Timeout** parameter in the *tunneller.ini* file) in case of communication failure.

By default, on Tunneller SSC the number of connections to the end OPC server for each combination of ***Tunneller CSC IP address/OPC Server ProgID/Category (A&E, DA or HDA)*** (starting in version 3.1.0.0, additionally **local Domain\User**) is restricted to ***1***. Starting in version 3.0.2.0, this restriction can be turned off by setting the **DeleteDuplicateSessionsOnNewConnection** parameter to ***0*** (under the **TCConnection** section on *tunneller.ini* configuration file).

Connection Time, Timeouts, and Retries

Please refer to the **Connection Failure Scenario** section before reading this section.

Tunneller allows the user to set the **Processing Timeout**, **Communication Timeout**, and **Communications Retry** values using the **Client-Side Gateway Configuration Utility**. These values are used by Tunneller when sending and receiving data on the network.

Problem: Network Link is Broken

If the network link is broken, the time it takes Tunneller to detect the break depends on where the link broke. If it is the local connection, the detection will be quick. If the break occurs within the network (e.g., routers, switches), it may take Tunneller several additional seconds to detect it. The detection also depends on network usage. If the OPC client is not sending or receiving data for extended periods, Tunneller will not detect that the link is broken until communication resumes.

Note that if the **Use Remote Status** checkbox is selected, sending and receiving data will happen at least as frequently as the OPC client calls ***GetStatus***.

The value of the **Communication Timeout** parameter affects the network link failure detection time because it is the time the network will spend testing the link for the ability to perform the required operation (send/receive request/responses over the network). The longer the timeout value, the longer failure detection will take. A longer time also causes the responsiveness of Tunneller to go down because more time is spent testing the link.

The retry value is used when the network itself has problems but the link to the SSC is maintained. This could be because of a bad connection or interference caused by electrical equipment. If the network was unable to get the command to its destination because of a network problem, it will retry sending the command. However, many retry times are needed (i.e., up to the retry amount).

Notes:

- Tunneller versions prior to 3.1.0.0 can detect the network timeout state if the execution of operations on the remote OPC server takes a long time. For example, adding a large number of items.

  Starting in version 3.1.0.0, prolonged execution of operations by the OPC server does not cause communication timeout. If the operation execution takes too long, then SSC starts to send Keep-Alive messages letting CSC know that the communication channel is alright. CSC waits for **Processing Timeout** before detecting operation as failed.

- The **Communication Retry Attempts** parameter does not affect OPC calls to the remote OPC server. For example, if the remote OPC server returns an error result code for an update history call, additional attempts to update will not be made. The result will be returned to the OPC client as is.

  The AddItems call can be retried, but it is configured using different option**:** the **AddItemRetries** parameter defined in the *TunnellerOpts.ini* configuration file. A retry happens if adding items returns ***S_FALSE*** and error codes for all items indicate the process was unsuccessful.

Problem: Timeout Expired

When running under normal conditions, default timeout and retry values are acceptable. When an unusually long command is executed (e.g., reading a very large number of items or browsing very large address spaces on the end OPC server) and the **Processing Timeout** value is not high enough, the processing timeout can elapse before the command completes. If processing timeout elapses, CSC does not break the connection, but returns the error to the OPC client. When **Communication Timeout** elapses, the network link will be considered failed and Tunneller will start the reconnect procedure. If reconnection is not established during the time defined by **ReconnectTime** parameter (set in *TunnellerOPts.ini* file on CSC), or during **Communication Timeout** (in the case when **ReconnectTime** is less than **Communication Timeout**), then the command is also considered failed.

The **Processing Timeout** value must be set to allow the longest expected command.

The **Communication Timeout** can come into effect when the command being sent is very long and the network has a low ability to transfer data.

The **Communication Retry Attempts** parameter can come into effect when network communication is not reliable. Increasing the retry amount allows the network link to continue when otherwise it would have been considered as failed.

If CSC detects a **Processing Timeout** for the operation, but SSC later completes it and CSC receives a response with results, the message (level 2) will be printed on a log file: *TIMEOUT WARNING: Probably the processing timeout parameter should be increased by **X** seconds,* where ***X*** is an estimated increase value.

Note: If processing timeout occurs while running an add items request, by using the **MaxAddMessage** parameter (configured in the *TunnellerOpts.ini* file) the request can be split into sub-requests with a smaller amount of items added at once, so a lesser **Processing Timeout** value can be used.

Even if the OPC client adds items by small portions so that each initial add item request does not take a long time, this option can be important if the total number of added items is considerable, when the Client-Side Component establishes reconnection to the Server-Side Component without disconnection of the OPC client. If the connection between the Client-Side and Server-Side components is closed due to some reason (e.g., the Server-Side Component is restarted, or there was a network disconnection), items must be added again to the end OPC server.

Handling Shutdown Request from Remote OPC Server

Note: This section is relevant for Tunneller SSC version 3.2.0.0 or higher.

When a remote OPC server sends a Shutdown Request to the SSC (which acts as an OPC client), then SSC performs the following steps:

1. Sends notification to the CSC, so the shutdown event can be tracked on the CSC log file.
2. If there is any call to the OPC server in process, it waits until it is finished (while Processing Timeout is not elapsed).
3. Disconnects from OPC server.

The connection to the OPC server is restored and items are re-added when the SSC receives the next request from the CSC. Reconnection occurs with certain delay after disconnection, which is 10 times the **ConnectDelay** defined by the CSC. In most cases this delay is enough for the remote OPC server to stop and be ready to start again.

Handling Items Momentarily Unavailable After Disconnection

Note: This section is relevant for Tunneller SSC version 3.5.0.0 or higher.

When a connection to the OPC server is restored (after a disconnection), the CSC will attempt to re-add the items. If the add operation fails, a certain number of retries are performed. The number of retries are determined by the **AddItemRetries** parameter (in the *TunnellerOpts.ini* file, the default is set to one retry). If all of the retries fail, or if the items are partially added (some items are added and some are not), then the CSC will not perform any more attempts.

To make the CSC periodically attempt to add the pending items, set the **AddPendingItemsPeriod** parameter to a specific period in milliseconds (in the *TunnellerOpts.ini* file, by default it is set to ***0*** which disables the feature). **It is recommended that you set this period to a value that is large enough to avoid eventual impact on the ongoing communication.**

Note: When upgrading from a version prior to 3.5.0.0, the **AddPendingItemsPeriod** parameter may or not be present in the *TunnellerOpts.ini* file. If the parameter is not present in the options file, it can be added manually, as follows:

1. In the *TunnellerOpts.ini* file, locate the section called **[TSAddItems]**.
2. Under the line **DelayAfterAddItems=0**, insert ***AddPendingItemsPeriod=some period in milliseconds (zero if you want this initially disabled)***.
3. Save the file and then restart the CSC service.

Tunneller with MatrikonOPC Redundancy Broker

When connecting MatrikonOPC Redundancy Broker (ORB) to Tunneller (**OPC Client** -> **ORB** -> **Tunneller** -> **OPC Server**), ORB's **Standby becomes primary after failover** option must be selected (for versions prior to ORB 2.2.0.0). Deselecting this option (for older ORB versions) will cause ORB to failover and fail back repeatedly.

Because ORB is connected to the local Tunneller client, ORB cannot detect when the OPC server on the end of the Tunnelled connection goes down. Therefore, ORB requires that failover conditions be configured through "watchdog tags" (i.e., advanced failover conditions) which will detect an unresponsive OPC server.

For more information regarding ORB, refer to ORB documentation.

Limitations

MatrikonOPC Tunneller has the following limitations:

1. The A&E Support starting in version 4.0.0 of MatrikonOPC Tunneller provides subscribe-only access to A&E Condition events. The client may subscribe to and receive conditional events, but cannot acknowledge them, request a refresh, or otherwise interact with the state of a condition.
2. A&E Support does not include the optional Area and Source browsing portions of the OPC Alarms and Events specification.
3. OPC DA 3.0 support is limited to the IOPCItemSamplingMgt interface. If multiple clients are accessing the same item, and have IOPCItemSamplingMgt in use, all clients must use the same IOPCItemSamplingMgt parameters. The last set of sampling rate and buffering parameters set apply to all of the clients.

Refer to the *MatrikonOPC Tunneller Release Notes* for known issues.

Troubleshooting

This section is intended to assist you by providing **licensing** information and **MatrikonOPC Support** contact information. Also addressed here are some of the most common problems encountered, and questions asked, while using this OPC server. Please check the following **Problems and Solutions** section before contacting the MatrikonOPC Support team.

Problems and Solutions

Using Pi OPC client

Problem:	How is the Pi OPC client to be used with MatrikonOPC servers, if applicable?
Solution:	When using the Pi OPC client with MatrikonOPC servers, please go into the OPC server **Advanced Options** setting and select the **Enable Mass Tag Adding** checkbox. This causes Pi to validate and add all tags in one group at a time rather than one tag at a time, resulting in significant improvements in time and network bandwidth used when initializing. Although this is highly recommended with any MatrikonOPC server, the difference is most noticeable when using products such as MatrikonOPC SCADA servers and Tunneller where each OPC operation has a high latency.

OPC client not responding when connected to Tunneller

Problem:	Why is my OPC client not responding when I am connected to Tunneller?
Solution:	Verify that you have a valid network connection. If the Tunneller CSC loses its network connection to the Tunneller SSC, your OPC client may not respond while Tunneller tries to re-establish the connection (this depends on the OPC client). Verify that the timeouts are set to a length of time that will allow the longest expected communication to complete. Often this is not set high enough causing Tunneller to keep re-trying commands.

GetStatus indicates OPC server is OK, but server had died

Problem:	My OPC client is calling ***GetStatus*** on Tunneller and it is telling me that the OPC server is OK, but the OPC server has died. Why am I still receiving a good status from Tunneller?
Solution:	Tunneller is set to use its own status instead of the remote OPC server's status. Set Tunneller to use either of the remote status options. When these are set, Tunneller will return a failed status if the end OPC server has failed or is unreachable.

Unable to browse remote OPC Server message displayed

Problem:	I see a message stating "*Unable to browse remote OPC Server*" when browsing.
Solution:	This can be caused by: • A firewall is stopping network traffic. • There is a physical disconnection between the two computers on the network. • The computer on the remote side is not powered on. • The SSC is not running on the remote computer.

- Tunneller is looking for the remote Tunneller on a port different than the one on which the remote Tunneller is configured.
- The remote OPC server has returned a browse error.
- Encryption settings on Client-Side Component do not match with settings on Server-Side Component.

More information regarding why a connection cannot be established can be found in the log file for the Client-Side Gateway Configuration tool (*C:\Program Files\Common Files\MatrikonOPC\Common\ ClientSideConfig.log*). It prints the error code returned by the WSAGetLastError function of the Windows Sockets Library (detailed description of error codes can be found in Windows Sockets documentation).

Note: Logging for the Client-Side Gateway Configuration Tool is always turned on using a high level. Therefore the user account, under which the Client-Side Gateway Configuration Tool is running, must have write access rights for the file *C:\Program Files\Common Files\MatrikonOPC\Common\ ClientSideConfig.log*.

Browsing does not return anything

Problem:	I get nothing returned when browsing.
Solution:	This can be caused by: • There is nothing in the end OPC server to browse. • The end OPC server does not support browsing. • A firewall is stopping network traffic. • There is a physical disconnection between the two computers on the network. • The computer on the remote side is not powered on. • The SSC is not running on the remote computer. • Tunneller is looking for the remote Tunneller on a port different than the one on which the remote Tunneller is configured. • The remote OPC server has returned a browse error. • A Tunneller timeout setting may need to be adjusted in the Tunneller SSC or Tunneller CSC.

AddGroup, *AddItems*, *ValidateItems* behave the same

Problem:	Why do ***AddGroup***, ***AddItems***, and ***ValidateItems*** behave the same?
Solution:	Tunneller does its best to limit the network traffic. When an OPC client makes the above calls, they are grouped together into one call across the network. If any one of the above calls fail in the end OPC server, it is returned as a general error in either the ***AddItems*** return code or the ***ValidateItems*** return code. To see the specific error look in the SSC log.

HDA Client used, but calls not successful

Problem:	I am using an HDA client with Tunneller, but my calls are not succeeding. What is the problem?

Solution:	It is possible that your timeouts are not high enough for the historical data access calls to succeed. This may be especially true if you are using a low-bandwidth network connection. Try adjusting the **Processing** and **Communication Timeout** periods or the number of retries in the **Client-Side Configuration Utility** for the particular connection. Disconnect and then reconnect to the server. **Note:** If **Processing Timeout** has elapsed, in most cases the message (log level 2) will be printed on a log file: *TIMEOUT WARNING: Probably the processing timeout parameter should be increased by %d seconds,* where *%d* is the estimated increase value.

Getting incorrect values of 0

Problem:	Why am I getting incorrect values of ***0*** when I first add items through Tunneller?
Solution:	When an item is first added to Tunneller there will be at least one scan interval time where it does not have a value supplied by the end OPC server. In this time, it may happen that the end OPC client requires an update of all the items that have been added. If this is the case, the item with the invalid value may be sent to the client. To not have Tunneller send any items that have never been updated, set the ***AllowInitialUpdate*** option in the *TunnellerOpts.ini* file to ***0***.

Fail to add items on first try

Problem:	Tunneller seems to fail adding items on the first try.
Solution:	What causes this is a timing difference between Tunneller and the end OPC server. Tunneller is often able to start faster than an OPC server. If this is the case, Tunneller may try to add items to an OPC server that is still in the start-up process and unable to accept an add request. There are a number of ways to alleviate this situation. First, try the ***PostConnectDelay*** option in the *Tunneller.in*i options file. Setting this value will cause a delay after connecting before any other operation can take place (including adding items). The second way is configuring Tunneller to retry adding items several times if it fails. In version 3.0.0.0, new configuration options are added to provide this functionality in the ***TSAddItems*** section of the *TunnellerOpts.ini* file: ***AddItemRetries*** – number of attempts to add items. Default = ***1***. ***AddItemDelay*** – delay between two attempts in milliseconds. Default ***= 100***. Another way to alleviate this situation is to have the Tunneller SSC service have a dependency on the OPC server. This will cause Tunneller to start after the end OPC server. The last option is to have an external batch file that starts the OPC server then waits an applicable amount of time before starting Tunneller. In version 3.0.0.0 the new configuration parameter, ***DelayAfterAddItems***, is added in the *TunnellerOpts.ini* file with a default value of ***0***.

Update of large amount of items fails

Problem:	Tunneller fails to update a large number of DA items.
Solution:	In the previous release of Tunneller, when the OPC client connects to the end OPC server through Tunneller and subscribes for updates on a large number of OPC DA

	items (for example 25000 items with an update rate of 1 second), the Private Bytes consumed by *TunnellerServer.exe* (Tunneller SSC) process grows continuously. At the same time, timestamps appear to fall behind compared to a directly connected client. In version 3.0.0.0, a new configurable option is added to handle this situation when CSC cannot process all update messages coming in from the Tunneller SSC (***MaxSizeOfQueue*** in ***Communication*** section of the *tunneller.ini* file on Server-Side Component). By default, its value is ***5***. When the number of messages to be sent from SSC to CSC in the queue of messages exceeds this maximum, Tunneller stores does not create new update message, but stores the latest updates for subscriptions in a cache. Note that there is no buffering of values and each update coming from the end OPC server rewrites the previous update. When the queue allows, it forms a new update message containing the latest updates that were not sent, and sends them. The final effect will be equivalent to the reducing of the update rate. As a result some intermediate update data will be missing.

Adding multiple items causes present items to go bad

Problem:	When new items are added, quality for items previously added for a short period of time become bad.
Solution:	Reason for items changing to bad, is that SSC can deactivate OPC Group before adding items and activate it after adding them. The new ***DeactivateGroupWhileAddingfOfItems*** parameter is added on the *tunneller.ini* file to control the activation of the group while adding items. If its value is set to ***0***, then this problem will not occur.

Tunneller does not work on Stratus box

Problem:	When switching from primary hardware to secondary hardware on a Stratus box during installation or during normal operation, Tunneller does not function properly.
Solution:	Re-install Tunneller using the **Custom** installation type and select the **Installing on a Stratus system** checkbox. That will disable checking for a hardware license key and the algorithm of software licensing will be adjusted taking into account the Stratus box features. **Notes:** • To provide correct functioning of software licensing, the generation of software license request and installation of the software license received from MatrikonOPC should be done when Stratus runs under the same active hardware configuration. • If support for hardware license key is required under Stratus, please contact MatrikonOPC Support.

Add Remote Tunneller Connection returns no OPC servers, or returns only part of OPC servers installed on Server-Side

Problem:	Attempting to add a new Tunneller connection from **Client-Side Gateway Configuration Tool** does not return a complete list of OPC servers installed on the Server-Side Component.
Solution:	Probably a restriction of the access to OPC servers is turned on in the Server-Side Component. Check to see if the given client access is granted for desired OPC servers.

Cannot connect to the tunnelled OPC server from certain computers/OPC clients

Problem:	Cannot connect to the tunnelled OPC server from certain computers/OPC clients.
Solution:	Probably user-impersonation functionality is turned on in the Server-Side Component. Check to see if the given remote user/proper local user is configured, using **Server-Side Gateway Configuration Tool**.

Tunneller returns different set of historical data when connection to remote OPC Server is made via Tunneller and start or end time is defined as relative time

Problem:	If start or end data is given in relative format for a historical read request, and the connection to a remote OPC HDA server is made via Tunneller, the Read History request returns a set of data different than it would be if returned at direct connection.
Solution:	The point is that time values given in relative format (for example, "NOW") are interpreted and converted by the Tunneller Client-Side Component. Therefore, if the system time on the Client-Side and the Server-Side is not synchronized, connection via Tunneller will cause a different data set to be returned compared to that of a direct connection between the OPC client and the remote OPC server.

Tunneller cannot be installed on NT4

Problem:	Tunneller cannot be installed on NT4. An error message window pops up informing that SHGetSpecialFolderPath function could not be found.
Solution:	The cause of this problem is that the *shell32.dll* file (by default, located in *C:\winnt\system32* folder) has an older version number than the required 4.71 or newer. Upgrade the *shell32.dll* file according to instructions published on **http://www-1.ibm.com/support/docview.wss?uid=swg21179367**. **Note:** A file named *ie4shlnt.cab* from the Microsoft Internet Explorer 4.0 installation package is required for upgrading. It can also be found within an install package for Microsoft Internet Explorer version 6.0 SP1, which can be downloaded from Microsoft's website using the following link: **http://www.microsoft.com/downloads/details.aspx?FamilyID=1E1550CB-5E5D-48F5-B02B-20B602228DE6&displaylang=en**

ORB fails over constantly when used with Tunneller

Problem:	Why does ORB keep failing over when I use Tunnelled servers?
Solution:	To get Tunneller to work well with ORB please use the following steps. **Note:** This is applicable for ORB versions prior to 3.1.2.0. **WARNING:** You are about to modify the registry. **This can be dangerous to your computer so do so at your own risk.** 1. Go to the registry (**Start** -> **Run,** type ***regedit***, and click **OK**). 2. Navigate to find: ***HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{85D83A0C-EC8C-4DD0-AAE6-1DA1558FEDD8}\Options*** 3. Find the value labelled ***EnableIOPCEventServer*** and change the value from ***0*** to ***1***.

OPC A&E client have problems when receiving A&E Condition events

Problem:	Why does an OPC A&E Client have difficulty handling A&E condition events from a server through MatrikonOPC Tunneller?
Solution:	MatrikonOPC Tunneller supports subscribe-only access to A&E Condition events. Some OPC A&E clients expect that they can access other optional features of the OPC A&E specification for Condition events when those events arrive, and do not properly handle the message returned to indicate those features are not available. In this case, it is recommended that the OPC A&E client application is configured to apply a filter to the A&E subscription that does not include the condition events. If it is not possible to configure the client application to such a filter, the following steps will configure the MatrikonOPC Tunneller SSC to override any subscription filter used, and disable the subscription to condition events. 1. Go to the installed directory for the MatrikonOPC Tunneller Server side component – typically **c:\Program Files\Matrikon\OPC\Tunneller\Server-Side Gateway**. 2. Open the *tunneller.ini* file in that folder in a text editor such as Notepad. 3. Find the section marked **[AlarmsAndEvents]**. 4. Make the setting **BlockConditionEvents = 1**. 5. Save the file and restart the OPC A&E client application. This option may be disabled by changing the option back to **BlockConditionEvents = 0.**

OPC DA server is disconnected frequently when there are no active subscribed groups or items

Problem:	Why does MatrikonOPC Tunneller periodically disconnect and reconnect to the DA server when there are no active OPC group subscriptions or items?
Solution:	Some OPC servers have been found to stop responding to OPC DA requests when they are reconfigured or other elements of their environment change. The MatrikonOPC Tunneller SSC monitors traffic to and from the CSC. When no traffic is exchanged within a period of time, the SSC attempts to check the connection to the source OPC DA server, and will disconnect and reconnect to the server to rebuild all of the group subscriptions in an effort to get the requested data moving again. This is an option that can be disabled by the following steps: 1. Go to the installed directory for the MatrikonOPC Tunneller Server side component – typically **c:\Program Files\Matrikon\OPC\Tunneller\Server-Side Gateway**. 2. Open the *tunneller.ini* file in that folder in a text editor such as Notepad. 3. Find the section marked **[TCRemotePROGID]**. 4. Make the setting **Disconnect CSC on Status Ping Timeout=0**. 5. Save the file and restart the MatrikonOPC Tunneller SSC: a. Open the **Services** control panel by choosing ***Run*** from the **Start** menu, and typing in ***services.msc*** and clicking on the **OK** button.

b. Locate the entry **MatrikonOPC Tunneller SSC**.

c. Right-click your mouse on the entry. Select **Restart** from the displayed menu.

This option may be re-enabled by changing the option back to **Disconnect CSC on Status Ping Timeout=1**, and restarting the SSC service.

OPC client detects the Client-Side Gateway as a DA 3.0 server and cannot access data through the DA 3.0 interfaces

Problem: Why does my OPC client detect and use Tunneller as a DA 3.0 server when it cannot access DA data using DA 3.0 interfaces?

Solution: MatrikonOPC Tunneller provides support for one OPC DA 3.0 interface, IOPCItemSamplingMgt, otherwise it is not DA 3.0 compliant. Some clients have been found to detect the IOPCItemSamplingMgt and treat the entire Tunneller product as a DA 3.0 compliant server.

The IOPCItemSamplingMgt interface can be disabled to prevent the client from identifying the client-side components as a DA 3.0 server:

1. Go to the registry (**Start** -> **Run**, type ***regedit***, click **OK**).
2. Navigate to find the following:

   HKEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\{85D83A0C-EC8C-4DD0-AAE6-1DA1558FEDD8}\Options on a 32-bit version of Windows

   Or,

   HKEY_LOCAL_MACHINE\SOFTWARE\Classes\Wow6432Node\CLSID\{85D83A0C-EC8C-4DD0-AAE6-1DA1558FEDD8}\Options on a 64-bit version of Windows.
3. Find ***TestOPCServerIntfDisable***. If the value is not present, it can be added by selecting **New** -> **String Value** from the **Edit** menu.
4. Set the value to ***8192***.
5. Re-start the MatrikonOPC Tunneller Client-Side Gateway service.

This option may be re-enabled by changing the ***TestOPCServerIntfDisable*** registry value to ***0*** and re-starting the MatrikonOPC Tunneller Client-Side Gateway service.

Note: This option applies to all tunnelled DA servers provided by the MatrikonOPC Tunneller Client-Side Gateway on this machine. When the ***IOPCItemSamplingMgt*** interface is disabled, that functionality is not available to any OPC client using any tunnelled DA server provided by the Tunneller CSC.

Communications problems when CSC or SSC is on Windows 7 or Windows Server 2008

Problem: Why does MatrikonOPC Tunneller experience difficulty connecting or maintaining a connection when one side of the tunnelled connection is hosted on a Windows 7 or Windows Server 2008 operating system?

Solution: Microsoft introduced TCP/IP stack optimizations in newer versions of the Windows OS which do not work well with MatrikonOPC Tunneller. These optimizations can be disabled as follows on either CSC or SSC host computers:

1. From the **Start** menu, locate the **Command Prompt** under **Programs**, **Accessories**.
2. In the command prompt, enter the following commands:

   netsh int tcp set global autotuninglevel=highlyrestricted

   netsh int tcp set global rss=disabled

3. Re-start the MatrikonOPC Tunneller services on that machine.

The settings can be restored to the operating system defaults by repeating steps 1 through 3, but using the following commands in step 2:

netsh int tcp set global autotuninglevel=normal

netsh int tcp set global rss=enabled

For more information about these options, consult the following Microsoft articles:

http://support.microsoft.com/kb/947239 - AutoTuningLevel

http://support.microsoft.com/kb/951037 - Receive Side Scaling (RSS)

Note: Starting with version 4.1.0 of Tunneller, the install offers to set these settings on affected operating systems automatically. These instructions may be used to set or reset the settings manually.

WARNING: These options impact all programs using TCP/IP networking on the host computer.

Search the *MatrikonOPC Support Knowledge Base* at **www.opcsupport.com** to find the answers to other commonly-asked MatrikonOPC Tunneller questions.

Licensing

Most MatrikonOPC products require that some form of licensing criteria be met for it to function correctly.

MatrikonOPC Tunneller supports both software **and** hardware licensing.

Starting with Tunneller version 3.2.0.0, a new software licensing API is in use. For all new installations, software licenses will be of a different type than the ones used in previous versions. For upgrades from older versions, the existing software license will be still valid and therefore re-licensing is not required.

Licensing information is described in detail within the *Licensing Procedures* document which accompanies the *MatrikonOPC Tunneller User's Manual.*

Contacting Support

The MatrikonOPC Customer Services department (**www.opcsupport.com**) is available 24 hours a day, seven days a week.

Contact MatrikonOPC Support using the information below, or send an email (**support@MatrikonOPC.com**).

For Monday to Friday **daytime support** requests, contact MatrikonOPC Support using the regional phone numbers provided in Table 10.

Region	Office Hours	Contact Information
North America UTC/GMT -7 hours (MST)	8:00 am-5:00 pm	**+1-877-OPC-4-ALL**
Europe /Africa * UTC/GMT +1 hours (CET)	9:00 am-5:00 pm	**+49-221-969-77-0** **(Request OPC Support)**
Australia/Asia * UTC/GMT +10 hours (AEST)	9:00 am-5:00 pm	**+61-2-4908-2198** **(Request OPC Support)**

* Toll-free regional numbers coming soon!

Table 10 - MatrikonOPC Support Regional Contact Information

For **after-hours support** in all regions, please use the following number. There is no extra charge from MatrikonOPC for calling their after-hours support number.

Region	Contact Information
All	+1-780-231-9480

Table 11 - After-Hours Support

OPC Compliance

MatrikonOPC Tunneller passes the automated certification test with the **OPC Foundation Compliance Test** utility for DA 2.05A and A&E 1.10.

Note: The MatrikonOPC Client-Side Component implements A&E, DA and HDA functionality on separate services: MatrikonOPC Tunneller A&E CSC, MatrikonOPC Tunneller CSC, and MatrikonOPC Tunneller HDA CSC.

For more information on OPC, view the documents listed below (as well as other OPC Specifications) at **http://www.opcfoundation.org**. MatrikonOPC supports the following interfaces:

- *OPC Overview 1.0*
- *OPC Common Definitions and Interfaces 1.0*
- *OPC Data Access Specification 2.05a*
- *OPC Data Access Specification 3.00*
- *OPC Historical Data Access Specification 1.2*
- *OPC Alarms and Events Specification 1.10*
- *OPC Security 1.00*

Common Interfaces

The server supports the mandatory functionality specified in *OPC Common Definitions and Interfaces*. The server supports the following locales for result code translation:

- US English (0x0409)
- System default (0x0800)
- User default (0x0400)
- Neutral (0x0000).

The server allows the client to set the client name for each connection. The server supports the shutdown event notification client-side interface.

Alarms and Events

This application supports the following A&E interfaces and methods:

- IOPCCommon
- IOPCEventServer
    - GetStatus
    - CreateEventSubscription
    - QueryAvailablefilters
    - QueryEventCategories
    - QueryEventAttributes
- IOPCEventSubscriptionMgt
    - SetFilter
    - GetFilter

- SetReturnedAttributes
- GetReturnedAttributes

This application supports **Simple**, **Conditional**, and **Tracking** events. Conditional events are subscribe only. The events will be received, but the Acknowledge, Refresh, State, and Condition Name browsing functions are not supported.

Data Access 3.0

- IOPCItemSamplingMgt

Historical Data Access

This application supports the following HDA interfaces and methods:

- IOPCCommon
- IOPCHDA_Server
- IOPCHDA_Browser
- IOPCHDA_SyncRead
  - ReadRaw
  - ReadProcessed
  - ReadAtTime
  - ReadAttribute
- IOPCHDA_SyncUpdate
  - QueryCapabilities
  - Insert
  - Replace
  - InsertReplace
  - DeleteRaw
  - DeleteAtTime
- IOPCHDA_AsyncRead
  - ReadRaw
  - ReadProcessed
  - ReadAtTime
  - ReadAttribute
  - Cancel
- IOPCHDA_AsyncUpdate
  - QueryCapabilities
  - Insert
  - Replace
  - InsertReplace

- DeleteRaw
- DeleteAtTime
- Cancel

Appendix A Standard Data Types

The Standard data types and their descriptions are listed in Table 12.

Hex	Dec	Data Type	Description
0000	0	VT_EMPTY	Default/Empty (nothing)
0002	2	VT_I2	2-byte signed integer
0003	3	VT_I4	4-byte signed integer
0004	4	VT_R4	4-byte (single-precision) real
0005	5	VT_R8	8-byte (double-precision) real
0006	6	VT_CY	Currency
0007	7	VT_DATE	Date
0008	8	VT_BSTR	Text (UNICODE)
000A	10	VT_ERROR	Error code
000B	11	VT_BOOL	Boolean (TRUE = -1, FALSE = 0)
0011	16	VT_I1	1-byte signed integer
0012	17	VT_UI1	1-byte unsigned integer
0013	18	VT_UI2	2-byte unsigned integer
0014	19	VT_UI4	4-byte unsigned integer
2002	8194	VT_ARRAY | VT_I2	Array of 2-byte signed integers
2003	8195	VT_ARRAY | VT_I4	Array of 4-byte signed integer
2004	8196	VT_ARRAY | VT_R4	Array of 4-byte (single-precision) real
2005	8197	VT_ARRAY | VT_R8	Array of 8-byte (double-precision) real
2006	8198	VT_ARRAY | VT_CY	Array of currency values
2007	8199	VT_ARRAY | VT_DATE	Array of dates
2008	8200	VT_ARRAY | VT_BSTR	Array of text values
200A	8202	VT_ARRAY | VT_ERROR	Array of error codes
200B	8203	VT_ARRAY | VT_BOOL	Array of Boolean values
2011	8208	VT_ARRAY | VT_I1	Array of 1-byte signed integers
2012	8209	VT_ARRAY | VT_UI1	Array of 1-byte unsigned integers
2013	8210	VT_ARRAY | VT_UI2	Array of 2-byte unsigned integers
2014*	8211	VT_ARRAY | VT_UI4	Array of 4-byte unsigned integers

Table 12 - Standard Data Types

* Indicates that the Array Data Type is not supported for Tunneller.

Appendix B Installation

Once the system requirements have been met, you are ready to install the software.

Note: As part of the installation process, the **MatrikonOPC Analyzer** tool is installed and used to detect the system settings that affect the use of this software. No information is communicated back to Matrikon. Information is stored on this system **only** for future use by MatrikonOPC Support to assist with troubleshooting, if required.

Tunneller should be installed on all machines where the user wishes to have communication between an OPC client and an OPC server. There are three parts to a Tunneller install:

1. The Tunneller SSC (Server-Side Component or Server-Side Gateway) is the component that will connect to the desired OPC server.
2. The Tunneller CSC (Client-Side Component or Client-Side Gateway) is the component to which the OPC client connects.
3. The Tunneller CCT (Client Config Tool) is used to configure the connection between Tunneller CSC and Tunneller SSC.

Notes:

- Installing Tunneller may require you to restart the computer. To limit the need for a restart, please limit the activity of existing OPC products (i.e., starting and stopping of OPC clients and servers).
- If a firewall is present and configured between the two computers using Tunneller to communicate, please ensure that the firewall will not block TCP communication on the desired port before continuing. The default port is ***21379***.
- In previous versions of Tunneller, a mixed installation option was available allowing an installation on a computer where Tunneller 1.x already exists and is not uninstalled. **Mixed installation is no longer available.** Please uninstall previous versions of Tunneller prior to installation of the current version; if not uninstalled, Tunneller 2.x versions will be overwritten.

To install the software:

1. Insert the MatrikonOPC Tunneller CD into the CD drive.
2. If the MatrikonOPC **Welcome** screen does not automatically appear, double-click the *MatrikonOPCTunneller.exe* file. The **MatrikonOPC Tunneller – InstallAware Wizard** appears and the **Welcome to MatrikonOPC Tunneller Setup** screen (Figure 22) appears.

   Note: The **Version** number located in the lower left corner indicates the version number of the software that is being installed. The text "X.X.X.X" will be replaced with the specific product version.

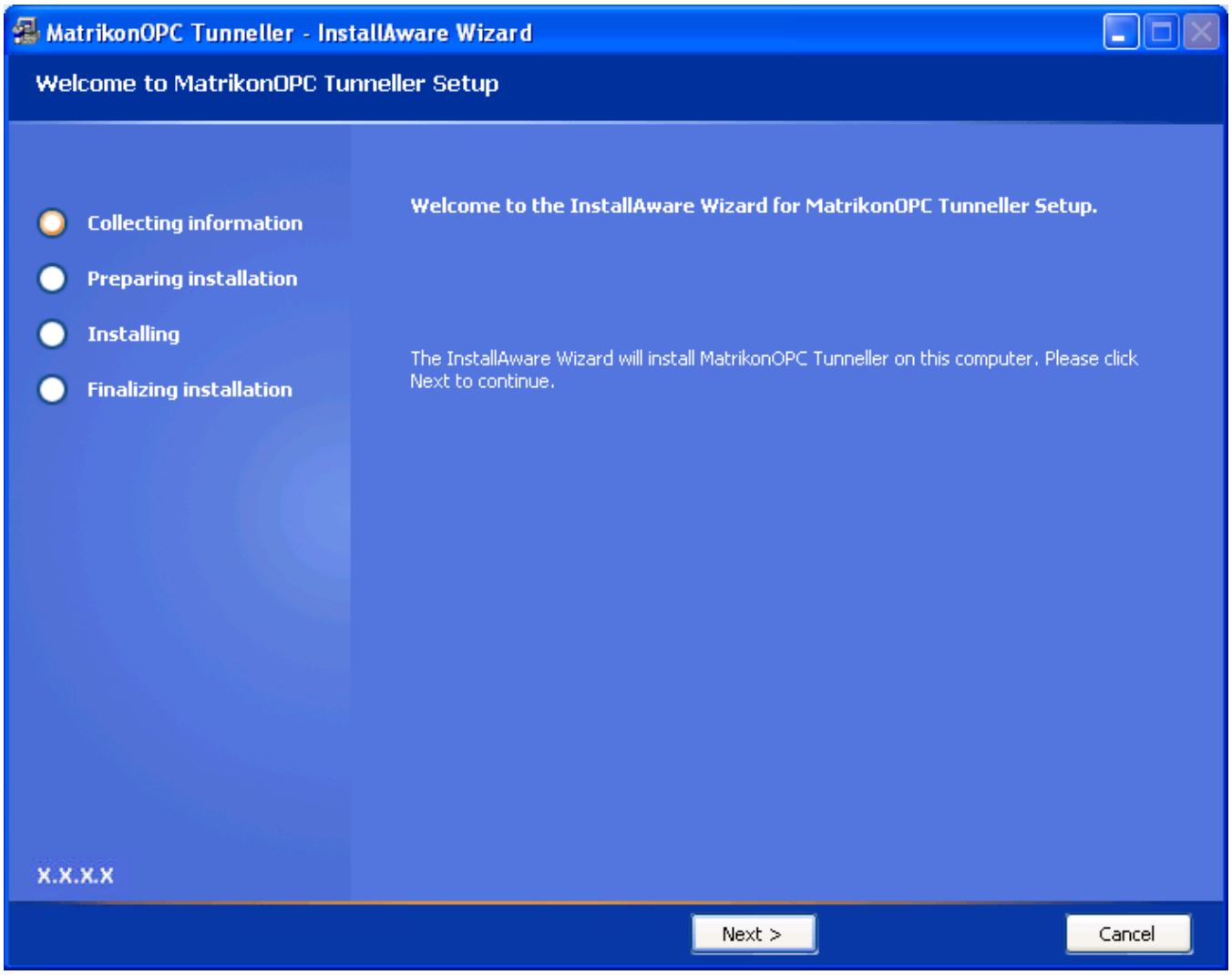


Figure 22 - Welcome to MatrikonOPC Tunneller Setup Screen

3. Click on the **Next** button.
4. The **License Agreement** screen (Figure 23) appears.

   Notes:

   - From the **License Agreement** screen, you have the option of clicking on the **Back** button to return to the **Welcome** screen, or selecting the **I reject the license agreement** option. Selecting the **I reject the license agreement** option button disables the **Next** button so your options are to return to the previous screen, cancel the install by clicking on the **Cancel** button, or select the **I accept the license agreement** option button enabling you to proceed through the install.
   - From this point onward, the **Back** button is available allowing you to return to the previous screen or screens.

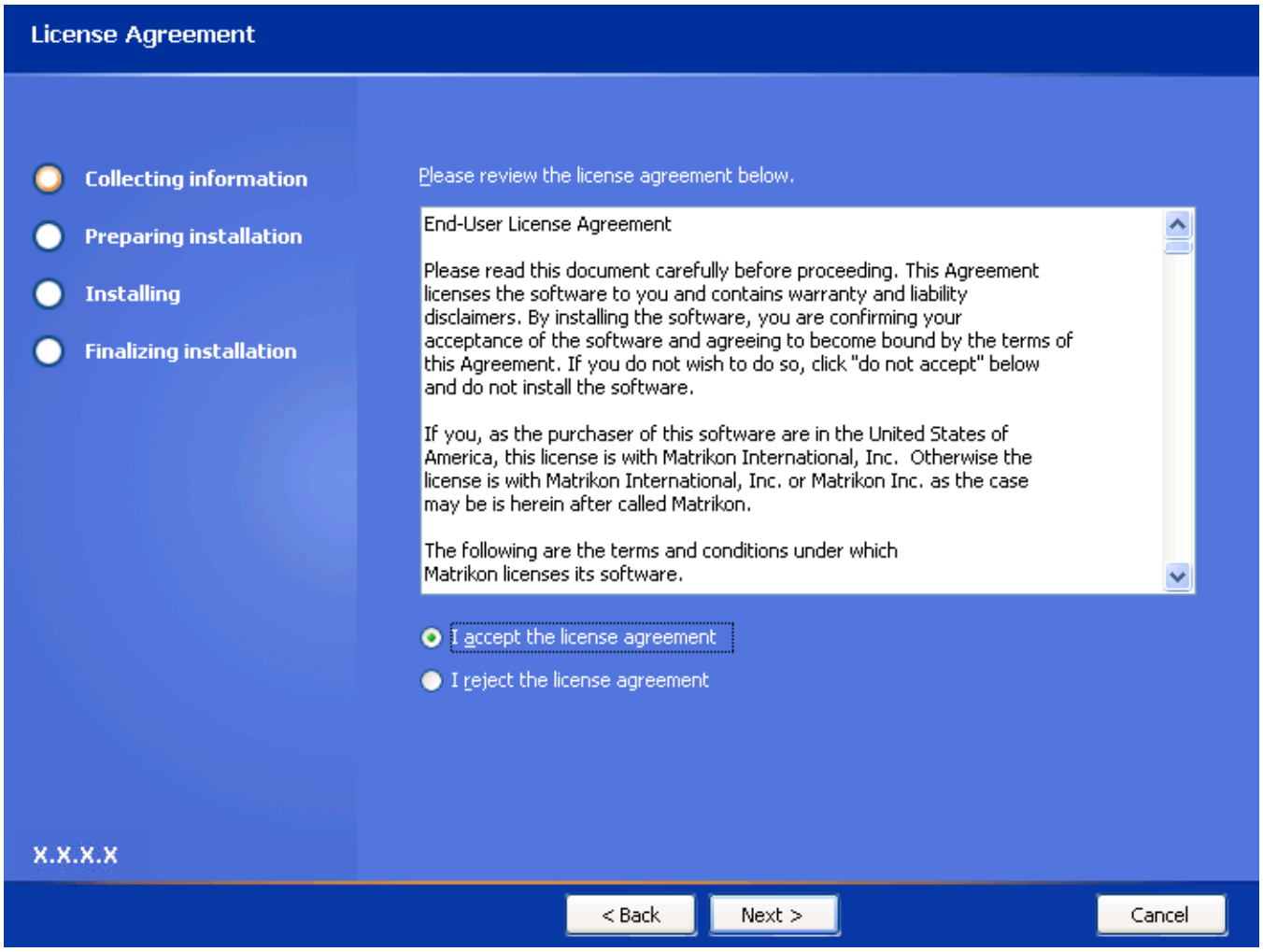


Figure 23 - License Agreement Screen

5. Read the **End-User License Agreement**, using the scroll bar to view the entire message.
6. Select the **I accept the license agreement** option button.
7. Click on the **Next** button. The **Setup Type** screen (Figure 24) appears.

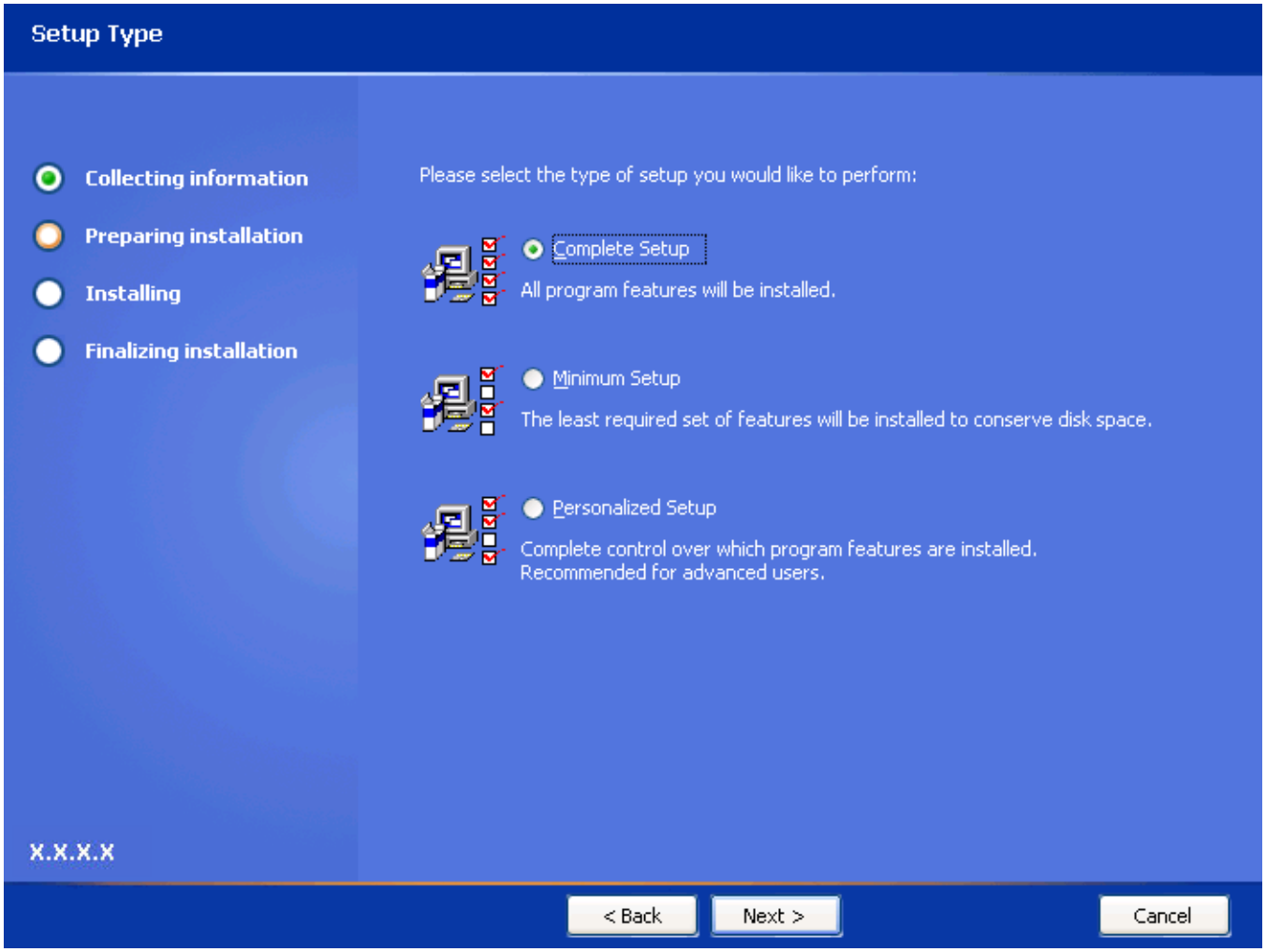


Figure 24 - Setup Type Screen

8. Select the type of setup to be performed.

   Note: Matrikon **recommends** that you select the **Complete Setup** option.

9. Click on the **Next** button. The **Destination Folder** screen (Figure 25) appears.

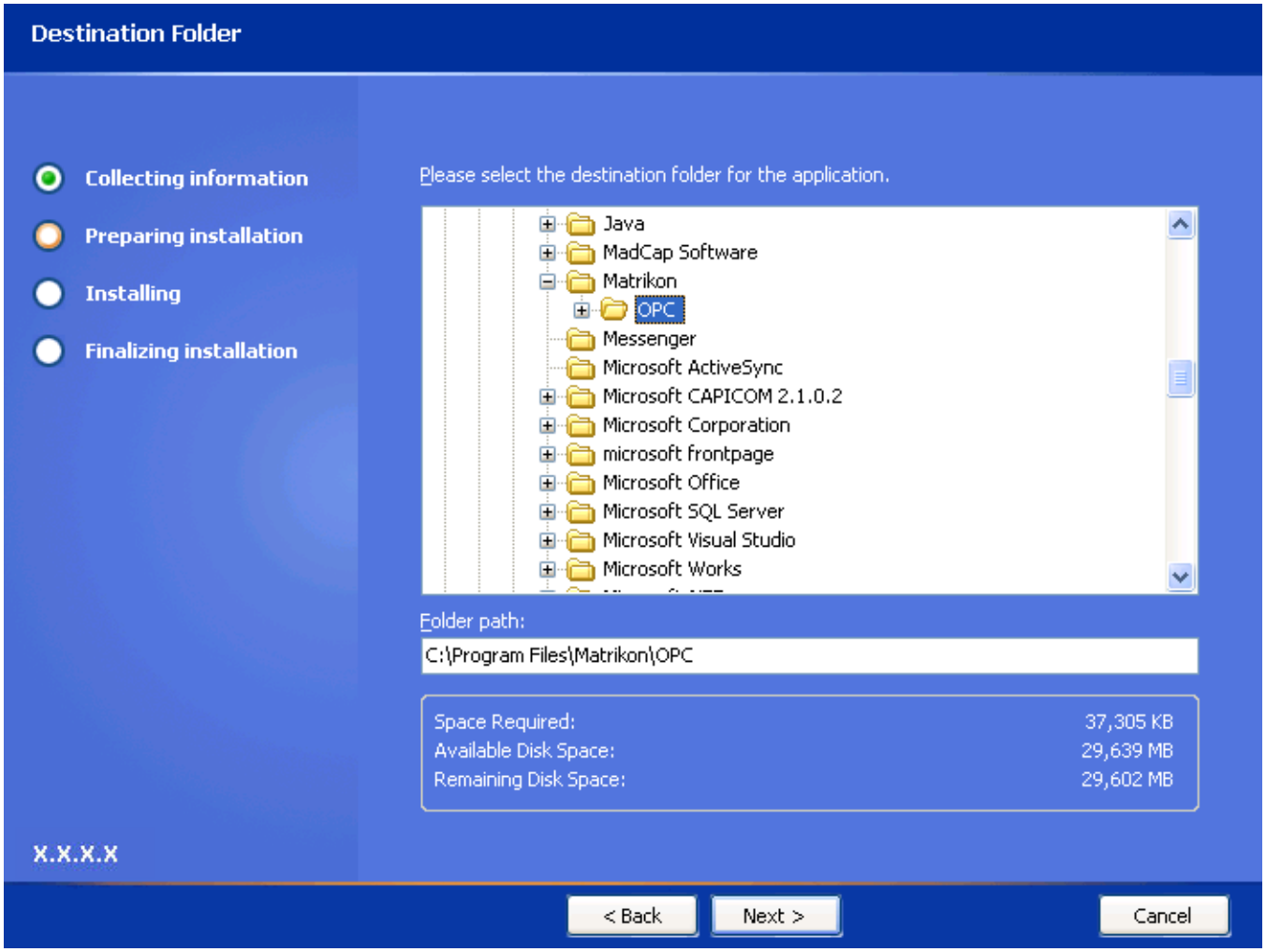


Figure 25 - Destination Folder Screen

10. Select the folder in which to install Tunneller.
11. Click on the **Next** button. The **Start Menu** screen (Figure 26) appears.

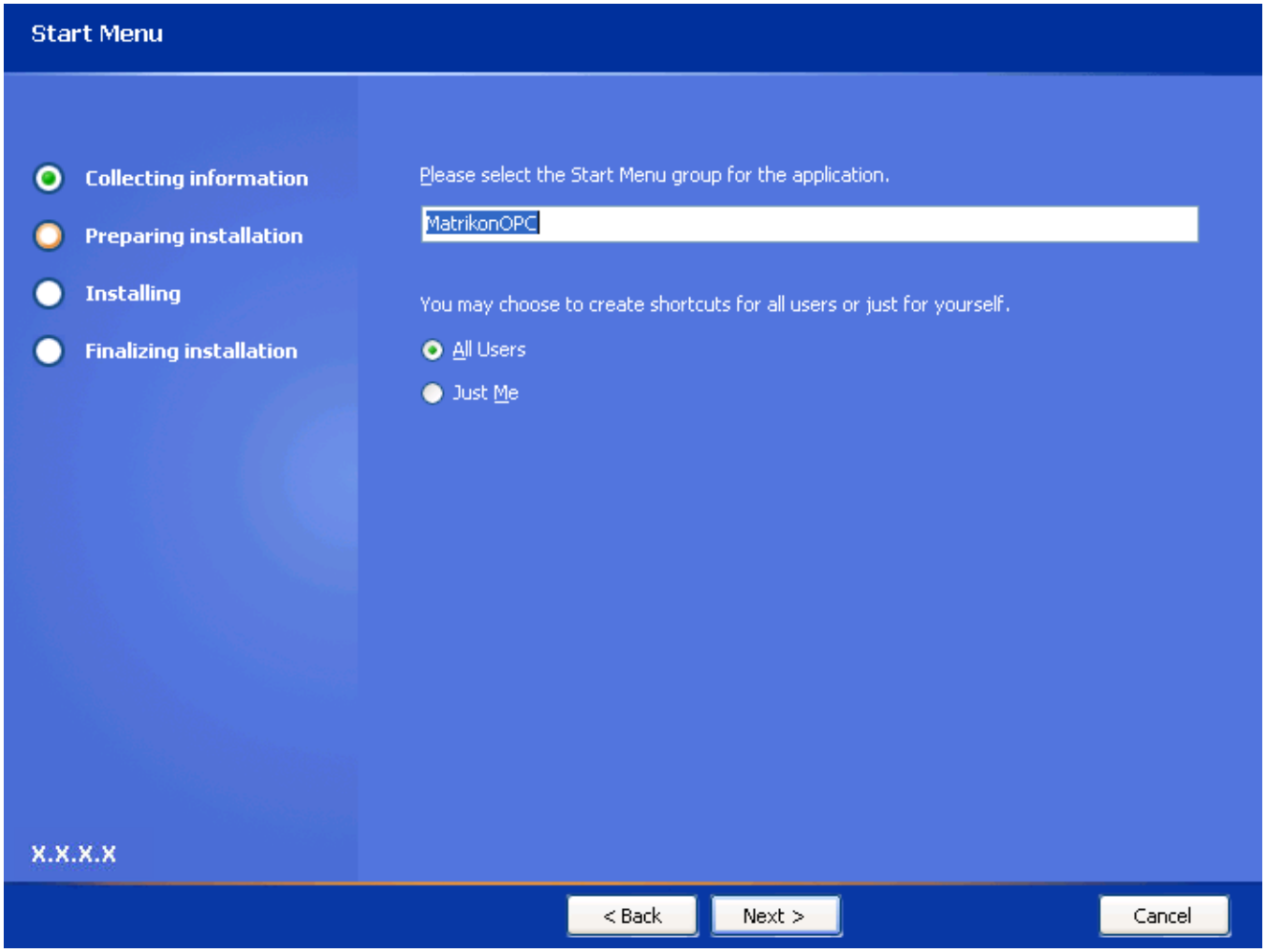


Figure 26 - Start Menu Screen

12. Select the **Start Menu** group and specify whether you want shortcuts created only for yourself, or for all users, by selecting the applicable option button.
13. Click on the **Next** button. The **Licensing** screen (Figure 27) appears.

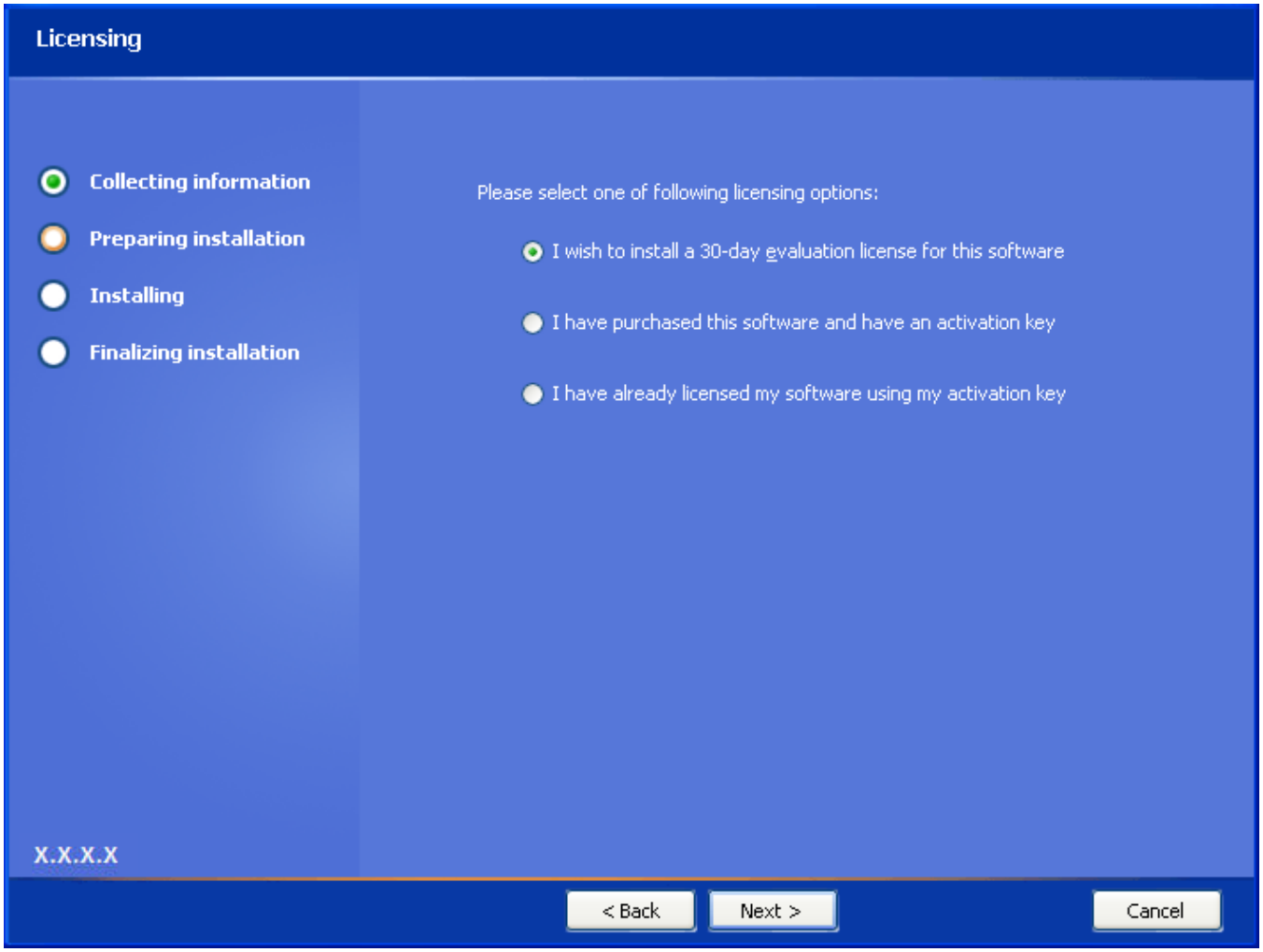


Figure 27 - Licensing Screen

14. Select the applicable licensing option.
15. Click on the **Next** button.
16. If DeltaV OPC Server is installed, a **DeltaV Admin** screen (Figure 28) appears. Otherwise, ignore this step and the next and go to step 20.

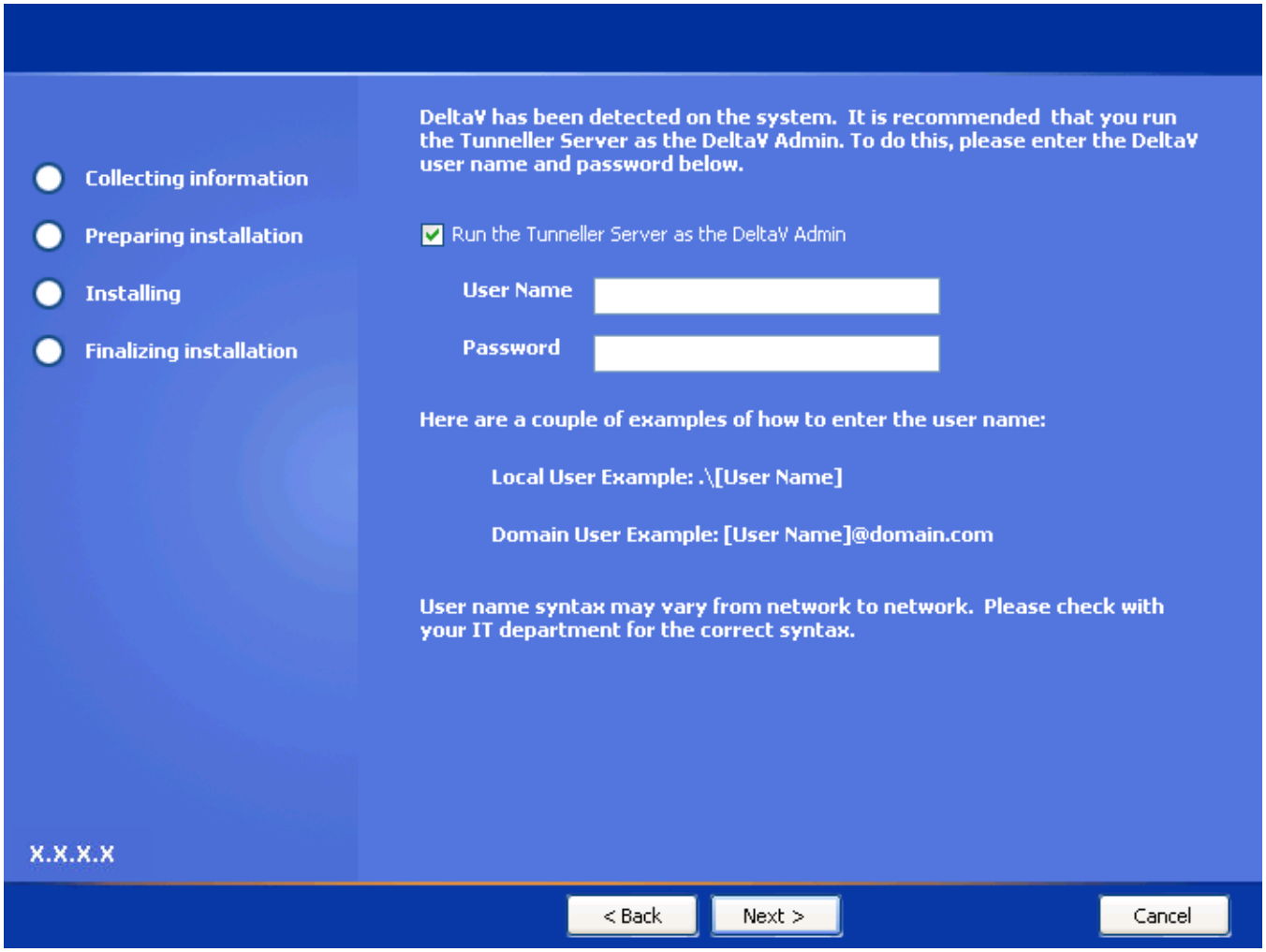


Figure 28 - DeltaV Admin Screen

17. If required, select the **Run the Tunneller Server as the DeltaV Admin** checkbox, and enter **DeltaV Admin** information (e.g., **User Name**, **Password**).
18. Click on the **Next** button. For Windows 7 or Windows Server 2008, the **TCP/IP Settings** screen (Figure 29) appears. If neither of those operating systems are detected, ignore this step and proceed to step 19.

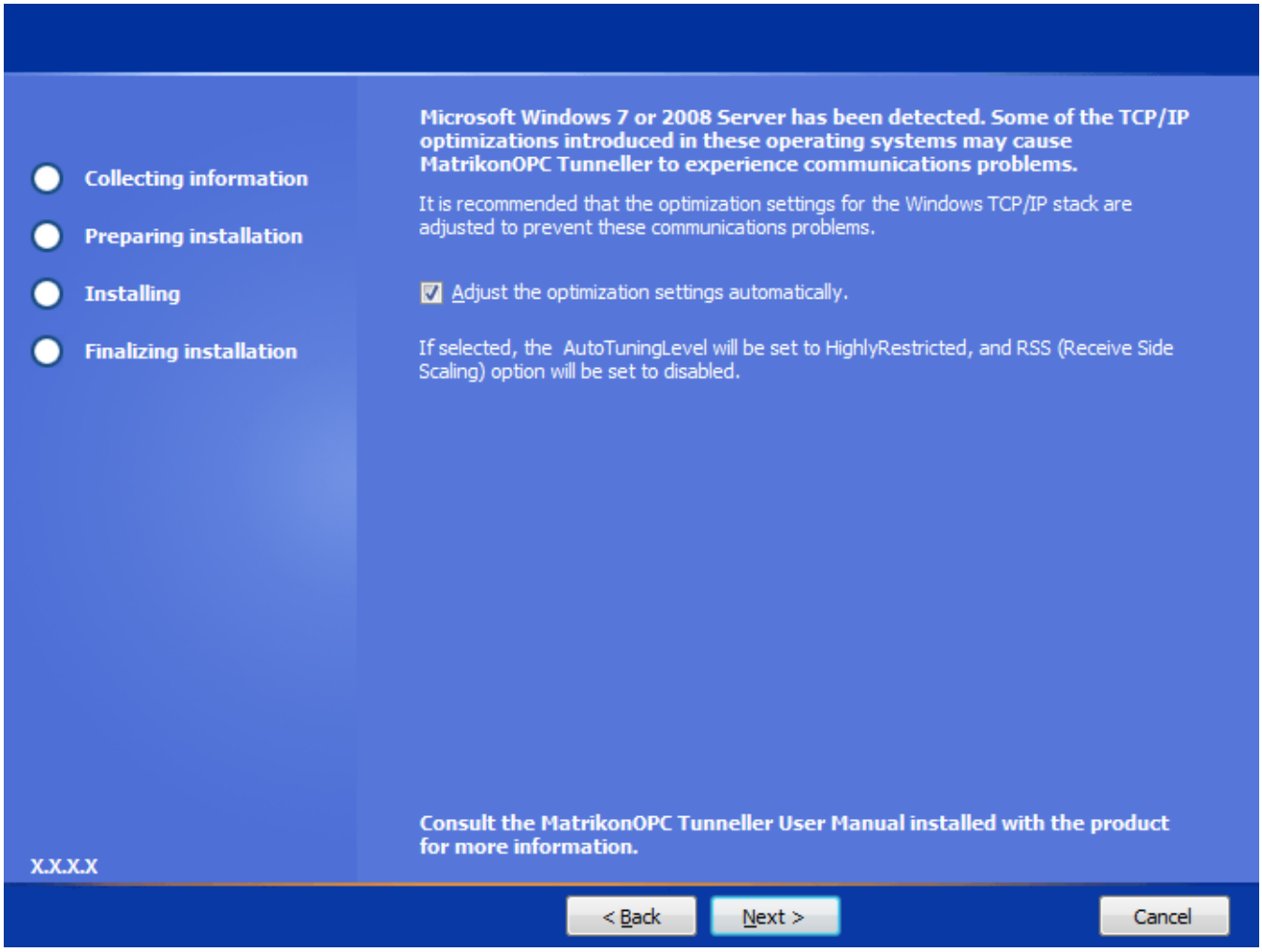


Figure 29 - TCP/IP Settings Screen

19. By default, the **Adjust the optimization settings automatically** checkbox is selected. If necessary, clear the checkbox to prevent the installer from adjusting Windows operating system TCP/IP settings. For more information, refer to the *Troubleshooting* section in this *User's Manual* (i.e., **Communications problems when CSC or SSC is on Windows 7 or Windows Server 2008**).
20. Click on the **Next** button. The **Ready to Install** screen (Figure 30) appears.

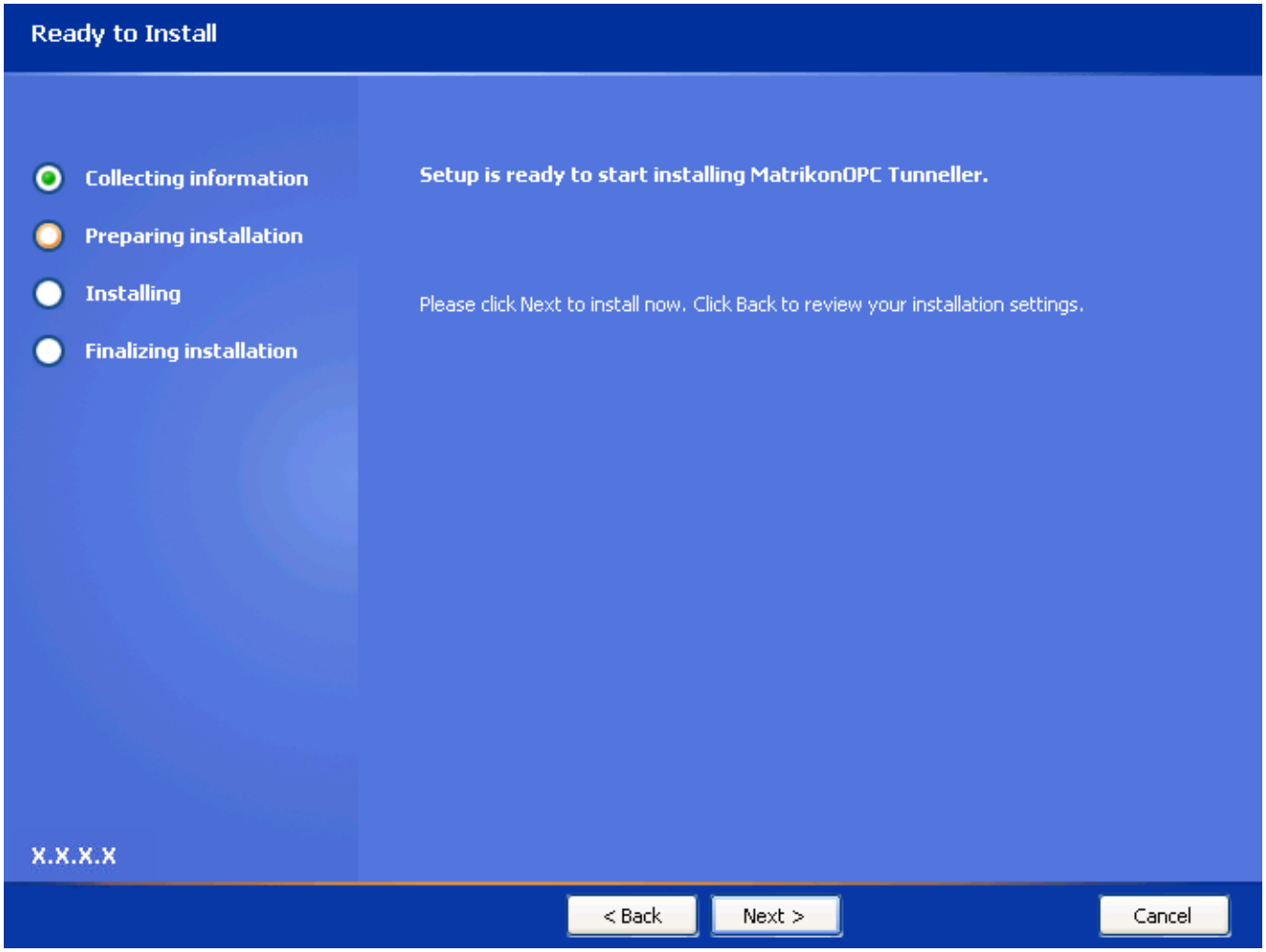


Figure 30 - Ready to Install Screen

21. Click on the **Next** button. The **Installing MatrikonOPC Tunneller** screen (Figure 31) appears, installation begins, and the product files are copied to the computer.

    Note: Prior to starting the installation, you have the option of clicking on the **Back** button to change any of the installation information. Click on the **Cancel** button if you wish to stop or cancel the installation.

Figure 31 - Installing MatrikonOPC Tunneller Screen

22. When the installation has finished, the **MatrikonOPC Tunneller Setup Complete** screen (Figure 32) appears stating that MatrikonOPC Tunneller has been successfully installed.

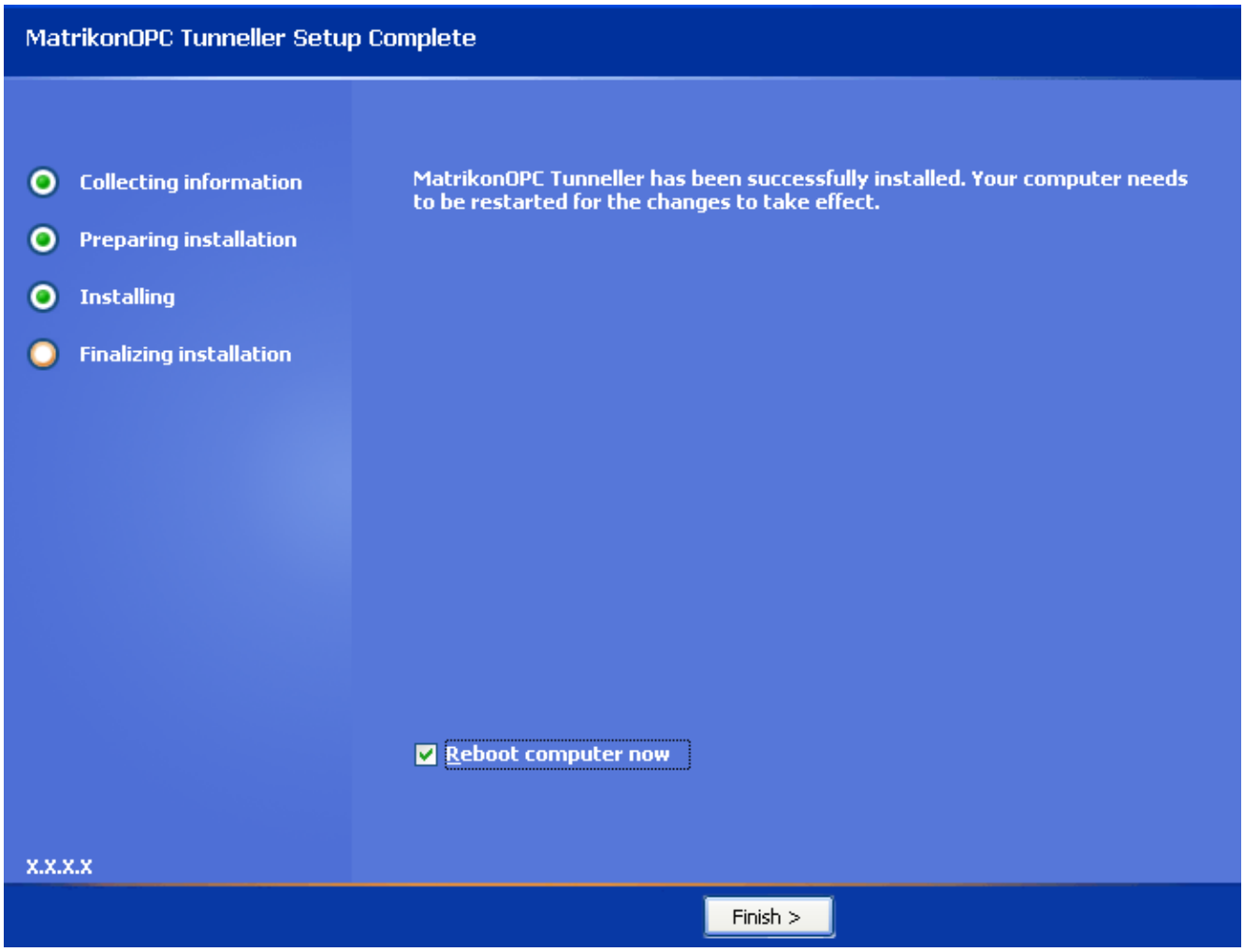


Figure 32 - MatrikonOPC Tunneller Setup Complete Screen

23. Click on the **Finish** button to complete the installation and exit the Wizard.

    Note: You have the option of restarting your computer now or later by selecting or clearing the **Reboot computer now** checkbox. If a reboot is not necessary, the following **Setup Complete** screen (Figure 33) appears. The checkboxes displayed depend whether Client-Side and/or Server-Side Gateway configurations are installed.

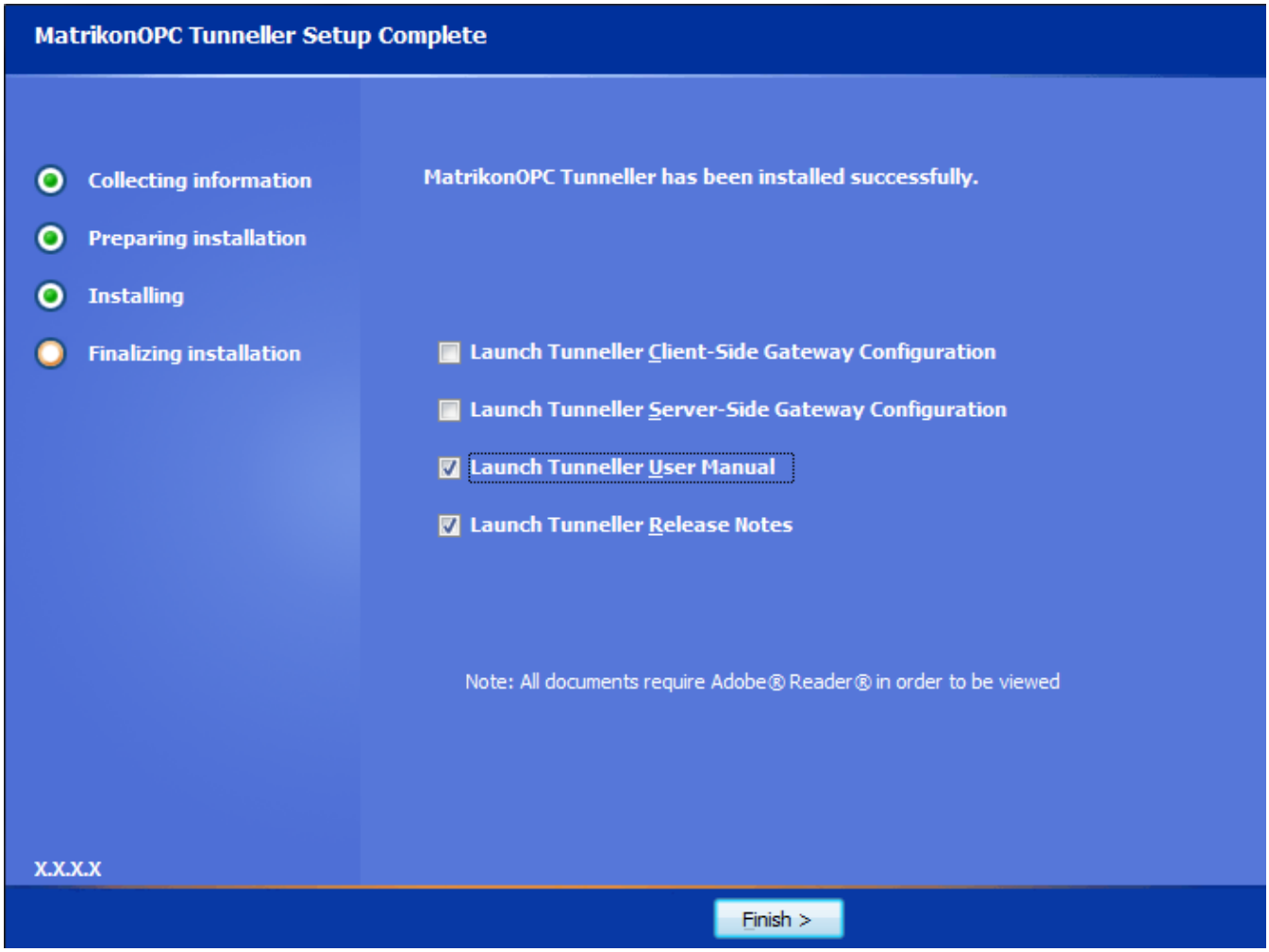


Figure 33 - MatrikonOPC Tunneller Setup Complete Screen (No Reboot)

24. The necessary files have now been copied to the target computer, the software components registered, and shortcut icons created in the **Start** menu.

Notes:

- If prompted to restart, the computer must be restarted prior to running MatrikonOPC Tunneller.
- If the Client-Side Gateway has been installed, access the icon in the **Start** menu program folder listing (**Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller** -> **Client-Side Gateway Config**) to configure Tunneller CSC.
- If the Server-Side Gateway has been installed, access the icon in the Start menu program folder listing (**Start** -> **Programs** -> **MatrikonOPC** -> **Tunneller** -> **Server-Side Gateway Configuration Tool**) to configure Tunneller SSC.

Note: At this point, it is recommended that you verify the DCOM settings. Reference to the DCOM configuration can be found in the **DCOM Manual**. This configuration varies for different operating systems.

Appendix C Installed Files

The installation program copies all necessary files to the target computer and creates shortcut icons in the **Start** menu.

The files listed in Table 13 are installed by default, in the following location:

C:\Program Files\Matrikon\OPC\Tunneller

File Name	Description
Client-Side Gateway	Folder containing files pertaining to the Client-Side Gateway (see Table 14).
Server-Side Gateway	Folder containing files pertaining to the Server-Side Gateway (see Table 15).
MatrikonOPC Tunneller User Manual.pdf	*User's Manual* for this product.
MatrikonOPC Tunneller AE	Tunneller A&E OPC server shortcut.
MatrikonOPC Tunneller HDA	Tunneller HDA OPC server shortcut.
MatrikonOPC Tunneller	Tunneller DA OPC server shortcut.
MatrikonOPC Tunneller Release Notes.pdf	*Release Notes* for this product.
Licensing Procedures.pdf	Licensing procedures document.
TunnellerOpts.ini	File containing Tunneller CSC options.

Table 13 - Files Installed in "Tunneller" Folder

The files listed in Table 14 are installed by default, in the following location:

C:\Program Files\Matrikon\OPC\Tunneller\Client-Side Gateway

File Name	Description
CCT.ini	Client-Side Configuration Tool options.
DSClientConfig.dll	Communications component of CCT.
lsapiw32.dll RmsApiProxy.dll	Licensing library files.
OPCTunneller.exe	DA Client-Side Component.
OPCTunnellerAE.exe	A&E Client-Side Component.
OPCTunnellerHDA.exe	HDA Client-Side Component.
PSTCFGTunnellerLib.ocx	Configuration panel for PSTCFG.
TunnellerClient.dll	In-proc class loader.
TunnellerClientConfig.exe	MatrikonOPC Tunneller Client-Side Gateway Configuration Utility.
EventLogger.dll	Contains event IDs for messages logged on Window Event Logger.
CSKeyManager.exe	Tool to configure encryption keys.
\Security\KeyFile.mkf	File contains encryption keys used in encrypted communication mode.

Table 14 - Files Installed in "Client-Side Gateway" Folder

The files listed in Table 15 are installed by default, in the following location:

C:\Program Files\Matrikon\OPC\Tunneller\Server-Side Gateway

File Name	Description
lsapiw32.dll RmsApiProxy.dll	Licensing library files.
tunneller.ini	Configuration file for the SSC.
TunnellerServer.exe	Server-Side Component executable file.
EventLogger.dll	Contains event IDs for messages logged on Window Event Logger.
SSKeyManager.exe	Server-Side Security configuration tool executable file.
tunneller.log	Server-Side Component log file.
\Security\KeyFile.mkf	File contains encryption keys used in encrypted communication mode.

Table 15 - Files Installed in "Server-Side Gateway" Folder

Note: Server-Side Security configuration tool and SSC create two additional files (*imp.conf* and *Conn.dat*) under the Security folder.

The files listed in Table 16 are installed by default, in the following location:

C:\Program Files\Common Files\MatrikonOPC\Common

File Name	Description
ACLConfig.exe	Security access list configuration tool.
ClientSideConfig.log	Client side configuration log.
EULA.pdf	EULA document (End-User License Agreement).
LicenseRemover.exe	License removing utility.
LicenseWizard.exe	Licensing wizard.
OEM_Matrikon_OPC.dll	MatrikonOPC OEM badge library.
OPCAuto.dll	MatrikonOPC Automation Component – enables developers to access OPC data from client applications developed using automation tools.
OPCDAAUTO.dll	MatrikonOPC DA Automation Component – enables developers to access OPC DA data from client applications developed using automation tools.
opcda20_auto.doc	MatrikonOPC Automation Component interface standard.
opchda_ps.dll	The proxy-stub files to allow OPC clients to make remote connections to an OPC HDA server.
opchda10_auto.doc	Developer documentation for the HDA Automation Component.
OPCHDAAuto.dll	MatrikonOPC HDA Automation Component – enables developers to access OPC HDA data from client applications developed using automation tools.
PSTCFG.exe	Matrikon product configuration utility.

File Name	Description
PSTCFGMatrikon.OPC.Tunneller.1.LOG	Tunneller DA OPC server log.
PSTCFGMatrikon.OPC.TunnellerAE.1.LOG	Tunneller A&E OPC server log.
PSTCFGMatrikon.OPC.TunnellerHDA.1.LOG	Tunneller HDA OPC server log.
PSTcfgps.dll	Matrikon product configuration marshalling library.

Table 16 - Files Installed in "MatrikonOPC\Common" Folder

Note: If either MatrikonOPC Simulation Server or MatrikonOPC Explorer is installed, refer to the *User's Manual* for these products for the list of additional installed files.

The files listed in Table 17 are installed in the ***WINDOWS\system32*** folder:

File Name	Description
EXPREVAL.DLL	Expression Evaluation Library for Alias Equations.
OPC_AEPS.DLL	OPC Alarms and Events 1.0 Interfaces Marshalling Library.
OPCBC_PS.DLL	OPC Batch Custom 2.00 Proxy/Stub Library.
OPCCOMN_PS.DLL	OPC Common Interfaces and Marshalling Library.
OPCDXPS.DLL	OPC Data eXchange 1.00 Proxy/Stub Library.
OPCENUM.EXE	OPC Server List Component.
OPCHDA_PS.dll	OPC Historical Data Access 1.20 Proxy/Stub Library.
OPCPROXY.DLL	OPC Data Access 2.0 and 1.0a Interfaces and Marshalling Library.
OPCSEC_PS.DLL	OPC Security 1.00 Proxy/Stub Library.
ACTXPRXY.DLL	ActiveX Interface Marshalling Library.

Table 17 - Files Installed in "system32" Folder

Appendix D Un-Installation

To successfully un-install MatrikonOPC Tunneller, using the **Add/Remove Programs** from the Microsoft Windows **Control Panel** is recommended.

To un-install MatrikonOPC Tunneller:

1. Click on the **Start** button and highlight the **Control Panel** item.
2. From the displayed menu, select **Add or Remove Programs**.
3. The **Add or Remove Programs** window is displayed.
4. Scroll through the list of currently installed programs and updates to find and select **MatrikonOPC Tunneller**.

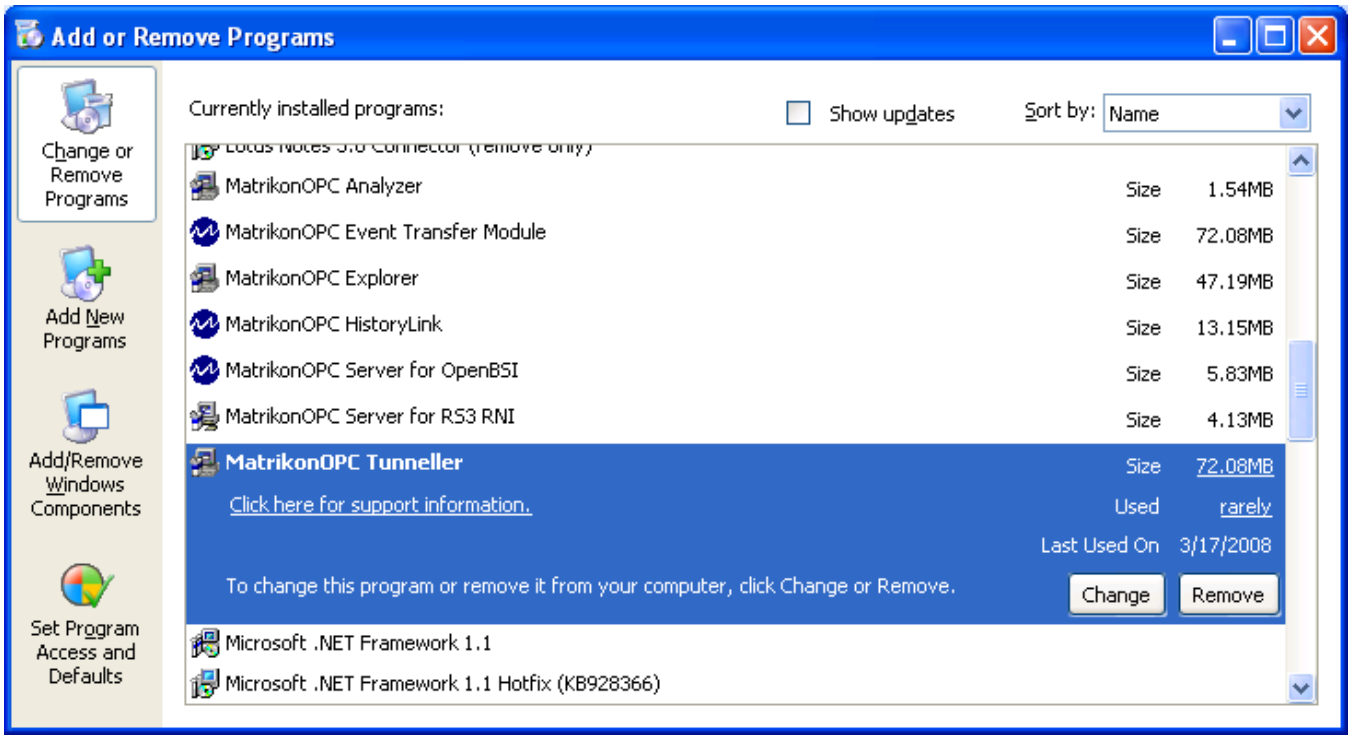


Figure 34 - Add/Remove Programs

5. Click on the **Remove** button associated with the MatrikonOPC Tunneller program to initiate the un-install process.
6. The **MatrikonOPC Explorer – InstallAware Wizard** appears and the **Welcome to MatrikonOPC Explorer Maintenance** screen (Figure 35) is displayed.

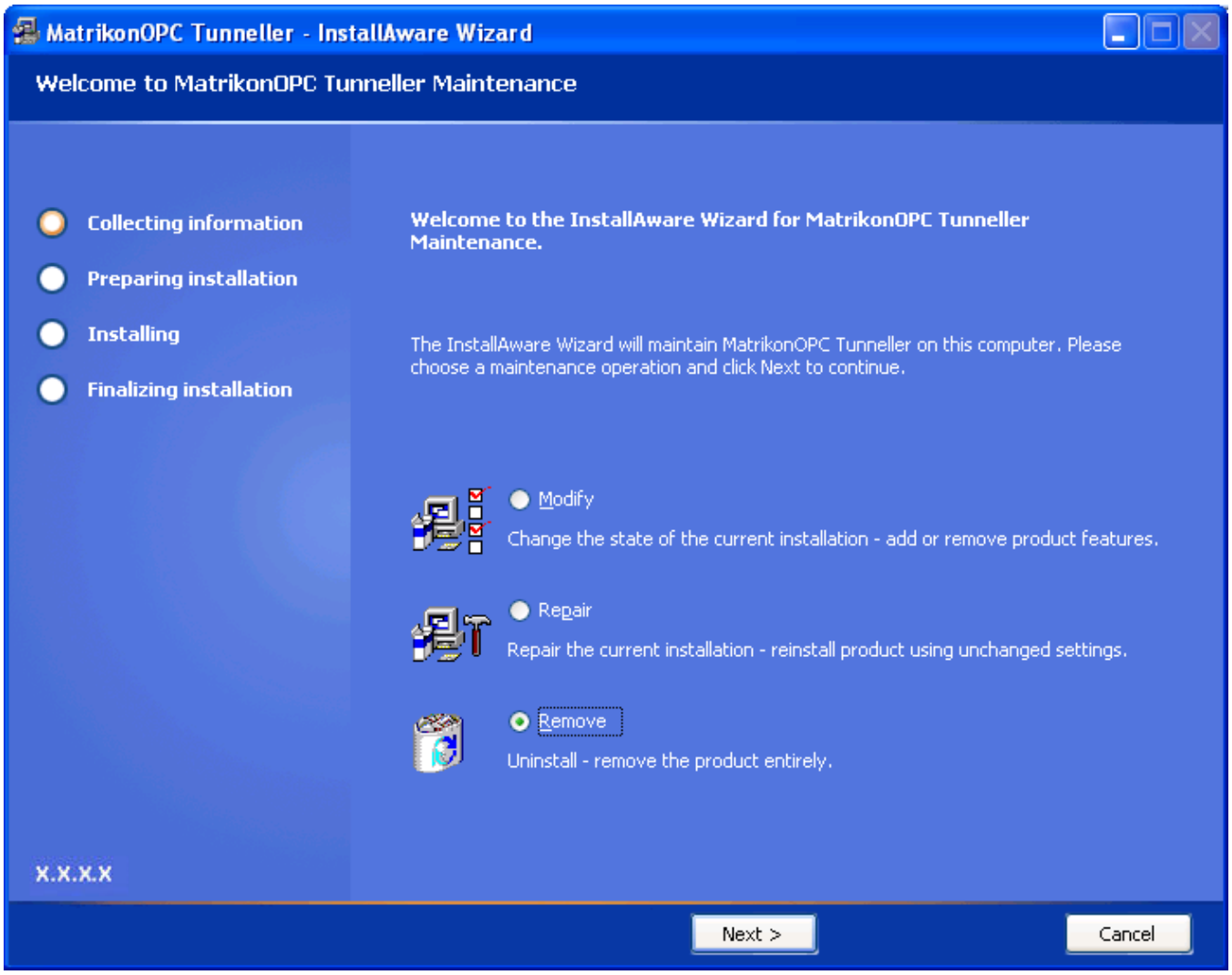


Figure 35 - Welcome to MatrikonOPC Tunneller Maintenance Screen

7. Select the **Remove** option button to un-install MatrikonOPC Tunneller entirely.
8. Click on the **Next** button.
9. The **Ready to Uninstall** screen (Figure 36) is displayed.

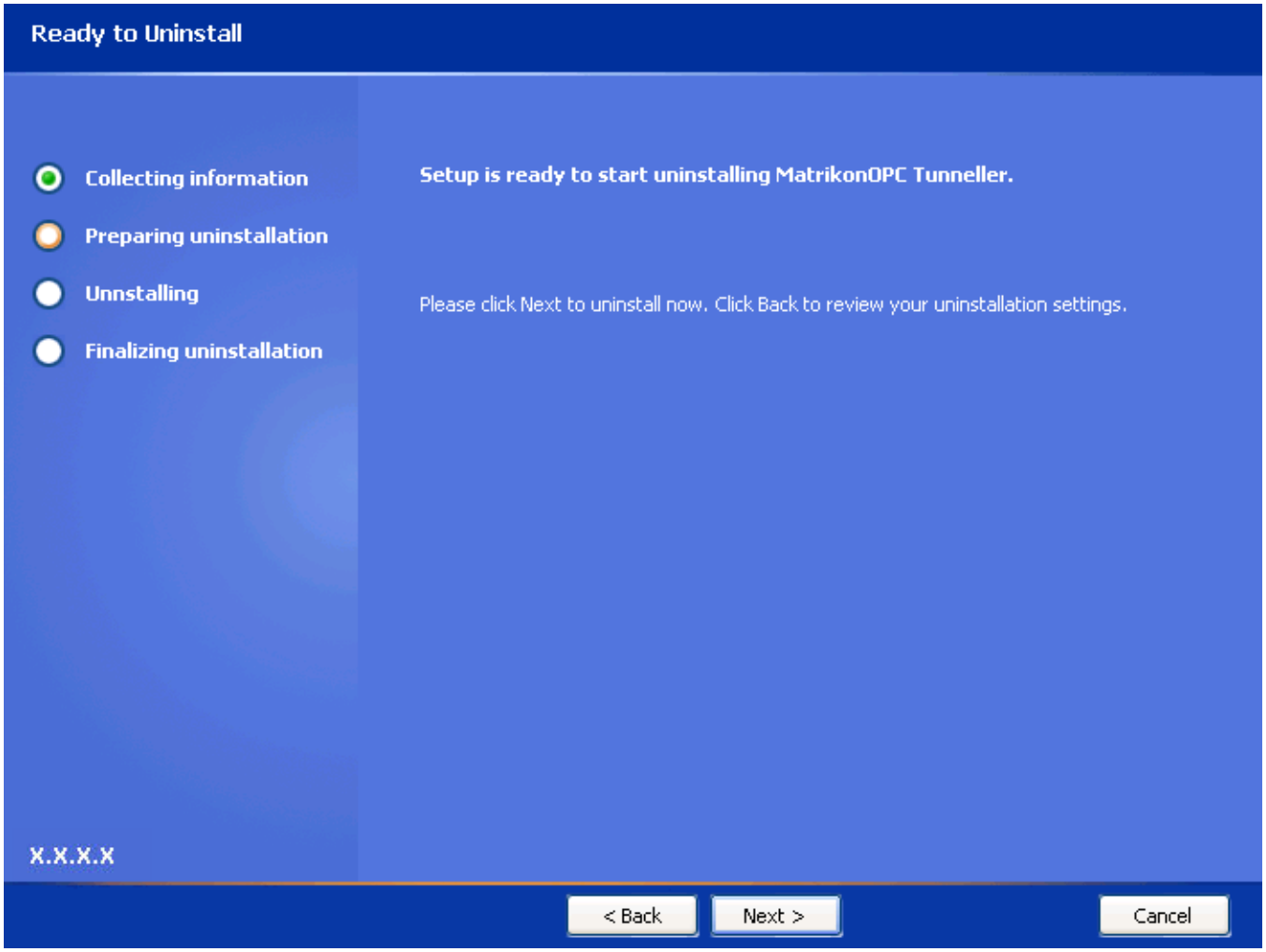


Figure 36 - Ready to Uninstall Screen

10. Click on the **Next** button.
11. The **Uninstalling MatrikonOPC Tunneller** screen (Figure 37) appears and the un-install takes place.

Figure 37 - Uninstalling MatrikonOPC Tunneller Screen

12. When the un-install has finished, the **MatrikonOPC Tunneller Setup Complete** screen (Figure 38) appears stating that MatrikonOPC Tunneller was successfully un-installed.

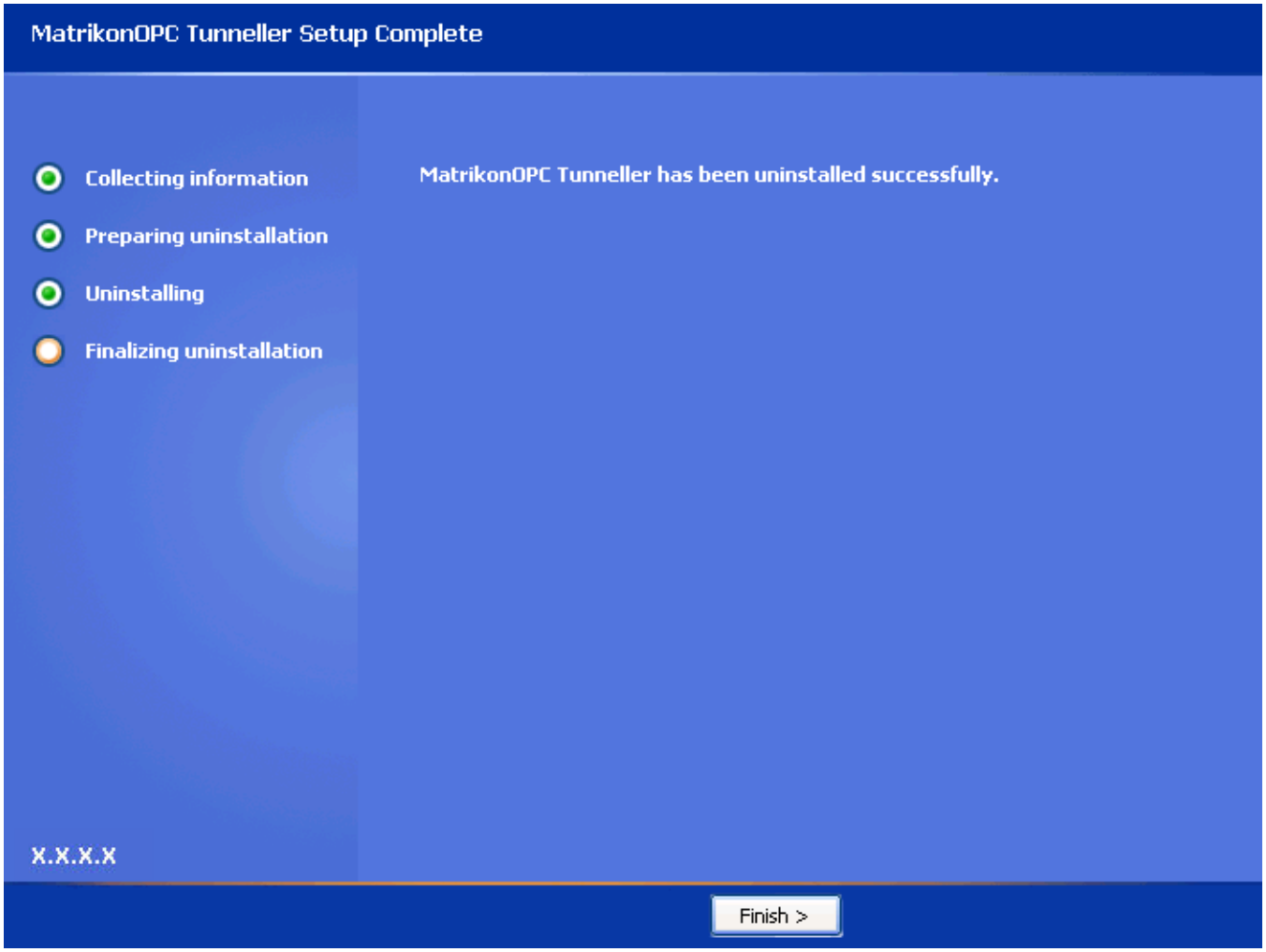


Figure 38 - MatrikonOPC Tunneller Setup Complete Screen

13. Click on the **Finish** button to complete the un-install and exit the Wizard.
14. The program no longer appears listed in the **Add or Remove Programs** window.