Server to Server Connector

- Name: **TcpDataAccess**
- Version: 1.18.0.3
- Protocol: Proprietary
- Interface: TCP/IP
- Runtime: .NET 2.0 (Multiplatfrom)
- Configuration:
  - Devices / Protocols

Channel Configuration
Node Configuration
Points Configuration
Data Access and Visibility

Overview

Communication Driver Information	
Driver name	TcpDataAccess
Assembly Name	T.ProtocolDriver.TcpDataAccess
Assembly Version	1.18.0.3
Available for Linux	True
Devices supported	Framework server applications compatible with the TcpDataAccess protocol
Manufacturer	N/A
Protocol	Proprietary over TCP/IP
PC Requirements	Ethernet port

Channel Configuration

Protocol Options

- **TagHistorian:** Defines if the historical data will be recovered from remote server.
- **TagProperties**: Defines the other tag properties beyond **Value**, **Quality** and **Timestamp**.

Node Configuration

Station Configuration

Stations syntax: *<Host> ; <Port>;[UserName];[UserPassword];[WindowsAuthentication]*

Where :

- ***<Host>***: IP Address of Server system
- ***<Port>***: TCP/IP port of Server system
- ***[UserName]***: User name with ClientStart permission in Server system. Optional parameter. Default value is the startup user.
- ***[UserPassword]***: Password for the user. Optional parameter. Default value is the startup user.
- ***[WindowsAuthentication]***: Created ***WindowsAuthentication*** option

E.g.: ***192.168.1.1; 3101***

Points Configuration

Address

The syntax for the TcpDataAccess communication points is: ***<ServerTagName>***

Where:

- ***<ServerTagName>:*** Tag Name in Server system.

E.g.: ***Tag.TagName***

> **AccessType**
>
> This driver requires the AcceptUnsolicited option to be enabled in order to read values from the server. Additionally, the ReadPollingRate parameter is only relevant for the initial read operation or for redundancy purposes.

Driver Revision History

TcpDataAccess Revision History	
Version	**Notes**
1.18.0.3	Points with a Backup Station no longer stay flagged Bad after a connection drop when the remote value does not change; polled access types now update their tags instead of going Bad and apply the Bit modifier; each link now reports its connection state, store-and-forward backlog and last flush time in the channel diagnostics; Tag Provider: browse the remote Asset tree and Alarm namespace, list top-level asset folders created without a parent, read tag properties reliably once they arrive, and run correctly on the Multiplatform (.NET 10) runtime; historian queries reuse the server connection.
1.18	Initial release on new documentation standards.

Data Access and Visibility

Not all Tags on the remote server can be accessible using this protocol. The visibility of the data and its access permissions, read or write, are defined by remote system, according to the Visibility property of the Tags in the remote system.

The available options are:

- **Private**: The tag remains visible only to the local project and its redundant pair. Use this setting to restrict the tag's visibility, ensuring that only the local project and its backup can access the tag. Private Tags will not be visible to remote Servers.
- **Protected**: The tag becomes visible to external connections for read-only operations. This setting allows external systems to read the tag's value without permitting any modifications, ensuring that the data remains unchanged. Protected Tags are accessible to read-only operations.
- **Public**: The tag becomes visible to external connections for both reading and writing. This setting provides full access to the tag, allowing external systems to both read and modify the tag's value. Public Tags are accessible to read or write operations.

In this section...
