Data Servers How-to

Step-by-step procedures for enabling and configuring Data Servers.

How-to Guides Data Foundation Data Servers How-to

Overview

Typical configuration scenarios for the UNS Data Server tools, with step by step instructions.

Overview
Enable MQTT Broker
Enable MCP for Runtime
Enable DataHub Service
Verify All Services
Troubleshooting
See Also

Enable OPC UA Server

Expose your UNS to external OPC UA clients.

Steps

1. Navigate to **Runtime Startup**
2. In **Startup Modules**, check **OPCServer**
3. Navigate to **Unified Namespace Data Servers**
4. Click the **OPC UA Server** edit icon
5. Configure settings:
    - **Port:** 4840 (default)
    - **Security Mode:** Select based on requirements
    - **Authentication:** Configure user access
6. Click **OK** to save
7. Restart solution for changes to take effect

Verify

1. Start the solution
2. Use an OPC UA client to connect to `opc.tcp://localhost:4840`
3. Browse the address space to see your UNS tags

Enable MQTT Broker

Provide publish/subscribe access for IoT devices and cloud platforms.

Steps

1. Navigate to **Runtime Startup**
2. In **Startup Modules**, check **MQTTBroker**
3. Navigate to **Unified Namespace Data Servers**
4. Click the **MQTT Broker** edit icon [??]
5. Configure settings:
    - **TCP Port:** 1883 (default)
    - **WebSocket Port:** 8083 (or 0 to disable)
    - **Authentication:** Configure as needed
    - **TLS:** Enable if required
6. Click **OK** to save
7. Restart solution for changes to take effect

Verify

1. Start the solution
2. Use an MQTT client to connect to `localhost:1883`
3. Subscribe to topics to see your UNS data

Enable MCP for Runtime

Allow AI models to query live solution data.

Steps

1. Navigate to **Solution Capabilities**
2. In the **Enable MCP for Runtime** group, check the box, then select the tool groups to expose: **UNS Tools**, **Alarm Tools**, **Historian Tools**, **Custom Tools**
3. (Optional) To add your own tools, navigate to **Scripts Classes** and create a new class with type **MCP Tool**
4. Define methods using the `[McpServerTool]` attribute:

```
[McpServerTool, Description("Returns current temperature")]
public string runtime_get_temperature(
    [Description("Full tag path, for example Tag.Temperature1")] string tag_path)
{
    return @Tag[tag_path].ToString();
}
```

Classes of type **MCP Tool** receive the `System.ComponentModel` and `ModelContextProtocol.Server` namespaces automatically — no `using` lines are needed. Use snake_case method names with a `runtime_` prefix so the tool is easy to tell apart from Designer tools in a mixed AI client configuration.

1. Save and build the solution
2. Start the solution
3. Configure your AI client (e.g., Claude Desktop) to connect

Configure Claude Desktop

Edit `claude_desktop_config.json`:

```
{
  "mcpServers": {
    "MySolution": {
      "command": "<ProductPath>\\fx-10\\net10.0\\RuntimeMCP.exe",
      "args": ["/host:127.0.0.1", "/port:<port>"],
      "transport": "stdio"
    }
  }
}
```

Verify

1. Restart Claude Desktop
2. Check **Settings Developer** shows "running"
3. In a new chat, verify solution appears in "Search and Tools"

See AI MCP for Runtime Connector for detailed setup See AI MCP for Runtime Tutorial for complete walkthrough

---

Enable DataHub Service

Synchronize data between multiple FrameworX solutions.

Steps (Publisher - Edge Solutions)

1. Navigate to **Runtime Startup**
2. In **Execution Environment**, check **Publish to a DataHub Service**
3. Click the gear icon [??] to configure
4. Set **Access Mode** to appropriate level
5. Click **OK** to save
6. Restart solution

Steps (Subscriber - UI Solution)

1. Navigate to **Runtime Startup**
2. In **Execution Environment**, check **Publish to a DataHub Service**
3. Click the gear icon [??] to configure
4. Set **Listening Port:** 30001 (default)
5. Configure **Access Key** if required
6. Click **OK** to save
7. Restart solution

Verify

1. Start all solutions (edge + UI)
2. Navigate to **Unified Namespace Services Monitor**
3. Verify DataHub connections show "Connected"

See Data Hub Communication Service for architecture details

Verify All Services

After enabling services, verify they're running:

1. Start the solution (**Runtime Startup Run Startup**)
2. Navigate to **Unified Namespace Services Monitor**
3. Check status for each enabled service:

Service	Expected Status
OPC UA Server	Running on port 4840
MQTT Broker	Running on port 1883
MCP for Runtime	Running (if Scripts enabled)
DataHub	Connected / Listening

Troubleshooting

Service Won't Start

- Check if port is already in use by another application
- Verify the service is checked in **Runtime Startup**
- Check Windows Firewall allows the port

External Client Can't Connect

- Verify the solution is running
- Check firewall rules for the port
- Verify authentication settings match client configuration
- For remote access, ensure IP/hostname is correct (not localhost)

MCP Not Showing in Claude

- Verify `claude_desktop_config.json` path is correct
- Restart Claude Desktop after config changes
- Check solution is running before starting Claude

See Also

- Data Servers — Concept overview
- UNS Data Servers Reference — All settings and options
- AI MCP for Runtime Connector — MCP details