# Runtime Diagnostics Reference

Diagnose runtime errors and bottlenecks.

**Reference** **Solution** **Runtime** UI | Engine | Troubleshooting | Diagnostics | Client-Server

---

Runtime Diagnostics (Reference): comprehensive diagnostic tools including Property Watch, Trace Window, and Module Information for monitoring and troubleshooting running solutions.

**Access:** Runtime  Diagnostics


Diagnostics UI Components
Configuration-Runtime Synchronization
Property Watch
Trace Window
Module Information
Performance Test
Synchronization Table Columns
Launching Diagnostic Tools
Best Practices Checklist


---

## Diagnostics UI Components

The Runtime Diagnostics interface consists of three main elements:

### 1. Module Status

Control panel with command buttons to Start, Stop, or Pause execution of any module.

### 2. Run Tools

Launch additional diagnostic tools:

- Property Watch
- Trace Window
- Module Information
- Performance Test

### 3. Runtime Connection State and Synchronization

Shows runtime connection status and whether Designer configuration is synchronized with runtime configuration in memory.

---

## Configuration-Runtime Synchronization

### Offline Configuration

In Offline Configuration mode, saved changes only update to runtime if Hot Reload is used. Otherwise, changes take effect at next solution start. Useful for making multiple changes without immediately affecting the running system.

### Online Configuration

With Designer connected to Runtime and online configuration enabled, all modifications apply immediately as you edit tables or save documents.

**Limitations:** Some operations cannot be applied online:

- Tag type modifications
- Client-Server domain changes for scripts
- Major structural changes requiring compilation
- Adding new Device Channels — the channel registers in the namespace immediately, but tag polling does not start until the next runtime restart. Stop and restart the runtime to bring the new channel into live communication.

These changes take effect at next system initialization.

### Synchronization Status

Common mismatch scenarios:

| Scenario | Cause | Resolution |
|---|---|---|
| Offline changes pending | Running application older than file | Hot Reload or restart |

| | | |
|---|---|---|
| Remote modifications | Another user modified runtime | Hot Reload to sync |
| Backup recovery | File restored but runtime unchanged | Hot Reload or restart |

### Hot Reload Command

Hot Reload applies updated configuration to runtime without shutdown, preserving tag values and objects in memory.

**Process:**

1. Identifies all modified objects (listed in synchronization table)
2. Briefly pauses affected modules
3. Applies changes to loaded modules
4. Resumes execution

---

## Property Watch

Tool designed to access and modify system tags and internal properties with IntelliSense support.

### Interface Elements

| Index | Description |
|---|---|
| 1 | Overlay window checkbox |
| 2 | Browser tab for object selection |
| 3 | Personalized monitoring list |
| 4 | Logged-in username display |
| 5 | Show all properties option |
| 6 | Selected object name |
| 7 | Module control panel |

### Usage

**Object Selection:**

- Type object name in Object column
- Use IntelliSense (type "tag." for suggestions)
- Select from browser tab
- Build custom monitoring list

**Value Modification:**

- Select property
- Enter new value (if permitted)
- Changes apply immediately

**Module Control:**

- Start/stop individual modules
- Monitor module status
- Test configuration changes

---

## Trace Window

System message display tool providing status information about system operations.

### Interface Elements

| Index | Description |
|---|---|
| 1 | Data header with filters |
| 2 | Message details |

| | |
|---|---|
| 3 | Auto-scroll option |
| 4 | Overlay checkbox |
| 5 | Settings window |
| 6 | Clear notifications |
| 7 | Stop/pause updates |
| 8 | Export (XML/CSV) |
| 9 | Close window |
| 10 | Error message details |

### Settings Configuration

**Message Types:**

- **Info**: User notices and status
- **Error**: Operation failures
- **Debug**: Developer information
- **Warning**: Potential issues

**Module Selection:** Choose specific modules to monitor for targeted diagnostics.

**Output Options:**

- Grid: Display in Trace Window
- File: Export to log file

**Object Monitoring:**

1. Enter object name in field
2. Select property from list
3. Click "Add" to monitor
4. Changes trigger trace messages

### Device Communication

Common use for finding device information:

- Read/write status
- TX frames (sent)
- RX frames (received)
- Invalid addresses (ControlLogix)

**Note:** For Devices, enable only error, info, and warning. Debug creates excessive data.

**CallStack Display:** Enable Debug in Settings and Debug Information in Runtime  Build and Publish.

---

## Module Information

Provides detailed operation information for all modules, including communication channels, reading groups, and error reports.

### Module Types

| Module | Information Displayed |
|---|---|
| **Alarm** | Alarm module specifics |
| **TCPServer** | Connected modules, clients, devices |
| **ScriptTaskClient** | Client task and expression statistics of a WPF Rich Client. Each connected Rich Client is listed with its own ID. |
| **Device Module** | Communication channel details |
| **Dataset** | Database access configurations |
| **Script Task Server** | Server expression/task statistics |

**ScriptTaskClient** is the client script module. Every WPF Rich Client starts it, whether or not **Allow Client Tasks (WPF only)** is checked in Solution Settings. That option only changes what you can set in the Designer. The module runs the built-in ClientStartup and ClientShutdown tasks and the Client-domain tasks and expressions, and it loads the Client-domain script classes. The HTML5 client does not start this module.

## Device Module Diagnostics

**Read Groups Information:**

- Virtual reading groups
- Runtime per item
- Successful/failed reads
- Error codes and timestamps

**Troubleshooting Steps:**

1. Check Read Groups Information for communication blocks
2. Systematic errors indicate access issues - verify node address
3. Block-specific errors - check tags and addresses
4. Use Trace Window for detailed error information
5. OPC discarded items show wrong addresses

## Controls

| Control | Function |
|---|---|
| Settings | Change display theme |
| Pause | Stop data updates |
| Close | Exit Module Information |

---

# Performance Test

Diagnostic feature to measure system performance through intensive resource usage simulation.

**Access:** Runtime  Diagnostics  Run Tools  Performance Test

## Test Methodology

The test executes memory-intensive operations:

csharp

```csharp
DateTime t = DateTime.Now;
for (int x = 0; x < 100; x++)
{
    int size = 1024 * 1024;
    int[] memAlloc = new int[size];
    for (int i = 0; i < size; i++)
    {
        memAlloc[i] = i;
    }
}
result = DateTime.Now - t;
```

## Performance Benchmarks

| Result | Assessment | Recommendation |
|---|---|---|
| < 300ms | Good | Suitable for most applications |
| 300-500ms | Acceptable | Monitor for large applications |
| > 500ms | Poor | Review system configuration |

**Virtual Machine Note:** VM configuration significantly impacts performance. Adjust VM settings if results are inadequate.

---

# Synchronization Table Columns

| Column | Description |
|---|---|
| **Module** | Specific module being monitored |
| **TableType** | Type of diagnostic table |
| **DesignerVersionID** | Designer module version |
| **RuntimeVersionID** | Runtime module version |
| **Designer Modified** | Last designer modification |
| **Runtime Modified** | Last runtime modification |
| **Description** | Module purpose and status |

---

## Launching Diagnostic Tools

Three methods to open diagnostic tools:

1. **Startup Window** (Server side)
    - Shows module status
    - Launch buttons for each tool
    - Tools run in separate windows
2. **Runtime Diagnostics Page**
    - Designer  Runtime  Diagnostics
    - Run Tools section
3. **Client Side**
    - Available from client applications
    - Remote diagnostics capability

**Note:** Tools may minimize to taskbar when launched.

---

## Best Practices Checklist

### Online Configuration

- Enable for real-time testing
- Monitor synchronization status
- Use Hot Reload for safe updates
- Test one module at a time

### Troubleshooting

- Start with Module Information for overview
- Use Trace Window for detailed errors
- Property Watch for specific values
- Performance Test for system capability

### Enterprise Test Mode

System only reads from field devices in test mode, even with write configuration present.

---

**In this section...**

---