

Media Level Redundancy Plug-In Help

© 2011 Kepware Technologies

Table of Contents

Table of Contents	2
Media Level Redundancy Plug-In Help	3
Overview	3
Redundancy Options	4
Switch On Failure	5
Monitor Items	6
Dynamic Changes to Redundancy Settings	6
Runtime Functions	7
Initialization	7
System Tags	7
Client Item Management	7
Error Descriptions	9
Event Log Messages	9
Item Validation Error Messages	9
'<text>' does not define a tag and is not a valid <driver name> address on <primary path>.....	9
'<text>' is not a valid <driver name> address on <secondary path>.....	9
Monitor item must specify a device tag.....	10
Monitor item specifies a tag that does not exist on the secondary device.....	10
Secondary device '<device path>' cannot be paired with primary '<device path>' because a circular dependency would resultSecondary device '<device path>' cannot be paired with primary '<device path>' because a circular dependency would result.....	10
Specified secondary device '<device path>' cannot be used because the driver type is not supported....	10
Specified secondary device '<device path>' is the same as the primary.....	10
Specified secondary device '<text>' does not exist.....	10
Index	11

Media Level Redundancy Plug-In Help

Help version 1.004

CONTENTS

[Overview](#)

What is the Media Level Redundancy Plug-In?

[Redundancy Options](#)

Where are the plug-in's settings configured?

[Runtime Functions](#)

What are the Media Level Redundancy Plug-In's Runtime functions?

[Error Descriptions](#)

What error messages does this plug-in produce?

Overview

The Media Level Redundancy Plug-In allows users that require device redundancy to maintain reliable communications to the critical hardware components being used in their systems. This server supports two types of redundancy: Communications Path Redundancy and Device Pair Redundancy. Descriptions of the types are as follows:

- **Communications Path Redundancy:** This redundancy type has the ability to communicate with the same device over multiple communications paths. For example, a device may support communications via TCP and RS-485. For redundancy, users can configure a primary communications path via TCP but allow the device to fall back to RS-485 when the Ethernet network is unavailable.
- **Device Pair Redundancy:** This redundancy type has the ability to communicate with two identical devices that are configured as a redundant unit (each running the same logic). For example, two redundant devices are on the same RS-485 network monitoring a process condition. Client monitoring software must be able to receive information from at least one of these devices in order to maintain the information display's integrity.

Note 1: To eliminate the communications path (and the device as the system's single point of failure), users can combine the redundancy types.

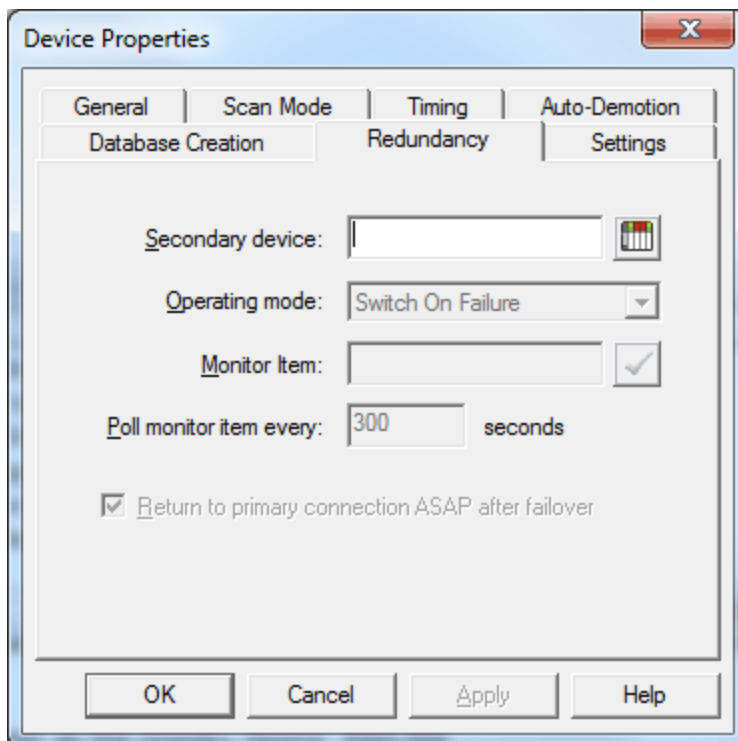
Note 2: A redundant connection contains a pair of devices that use Communications Path Redundancy or Device Pair Redundancy. The devices are designated primary or secondary. Redundancy configuration begins by identifying the primary device and proceeds by choosing a different, compatible device to use as the secondary device.

Supported Use

Media Level Redundancy is available on devices that communicate using serial or Ethernet drivers operating in master mode. Media Level Redundancy cannot be enabled if a suitable device does not exist for the secondary. Suitability is determined based on the following:

1. The device is not the same as the primary.
2. The device is not already in use as a primary in another redundant pair (such that a circular path back to the primary on the new redundant pair results).
3. The device uses a serial or Ethernet master driver for communications.

Redundancy Options



Description of the parameters are as follows:

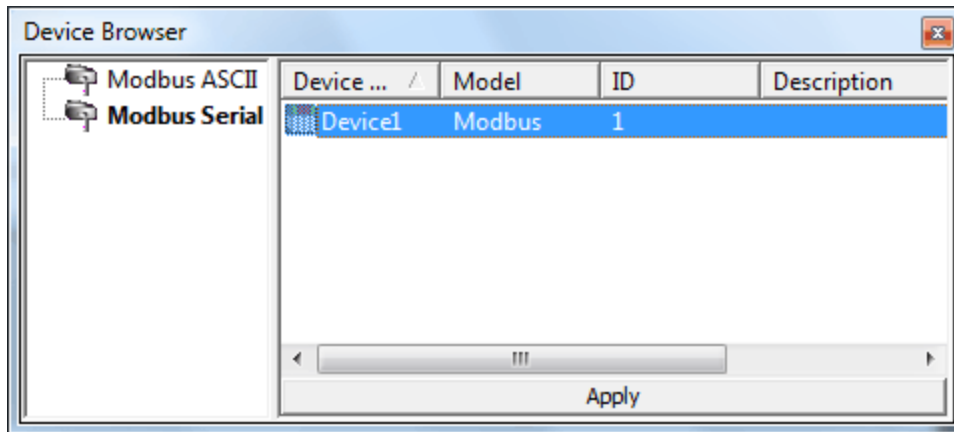
- **Secondary device:** This string specifies the explicit, unaliased path to the secondary device. The valid format is `<channel>.<device>`. The default setting is blank.
Note: If the device specified by this path is on the same channel as the primary device, the redundancy type will be Device Path Redundancy. Otherwise, it will be Communications Path Redundancy.
- **Operating mode:** This parameter specifies how the active device will be chosen at Runtime. Options include Switch On Failure, Primary Only, and Secondary Only. The default setting is Switch On Failure. Descriptions of the options are as follows:
 - **Switch On Failure:** In this mode, communications will fail over to the secondary device when the primary device enters an error state (and vice versa). Initial communications begin on the primary device. For more information, refer to [Switch On Failure](#).
 - **Primary Only:** In this mode, communications will be fixed to the primary device regardless of the device's error state. This option is used for maintenance when the secondary device needs to be taken offline.
 - **Secondary Only:** In this mode, communications will be fixed to the secondary device regardless of the device's error state. This option is used for maintenance when the primary device needs to be taken offline.
- **Monitor Item:** This optional parameter specifies a valid device tag name or address on both the primary and secondary devices that will be used to monitor communications on both devices. Valid formats include the device address (such as "400001") or a partially qualified tag name (such as "Tag1") to identify a Static Tag at the device level (or "Group1.Tag1" to identify a tag in "Group1" and so forth). The default setting is blank.
Note: This parameter is only available when the operating mode is set to Switch On Failure. Only tags that require device communications may be used. For more information, refer to [Monitor Items](#).
- **Poll monitor item every x seconds:** This parameter specifies the rate at which the Monitor Item will be polled on each of the two devices in a redundant pair. It is the rate at which the error tag will update on the inactive device in a pair. The valid range is 30 to 999 seconds. The default setting is 300 seconds.


Note: This parameter is only available when the operating mode is set to Switch On Failure.

- **Return to primary connection ASAP after failover:** When checked, this option indicates whether the primary device should become the active device as soon as possible after a failover. When checked, the primary will be re-activated as soon as possible following a transition of its error state from True to False.

Browsing for a Secondary Device

To browse for the device, simply click the icon located to the right of the entry field. In **Device Browser**, locate the desired device and either double-click or select **Apply**.



Note: A device with a redundant configuration will display the icon  in the server Runtime.

Switch On Failure

Failover

When the operating mode is set to Switch on Failure, failover to the secondary device will occur as soon as the primary device enters an error state. When the secondary device is active, failover to the primary will occur when the primary device is no longer in an error state.

The failed Read or Write that caused the error state to occur on the primary device will be submitted to the secondary device after switchover. Any subsequent failed Write Completion notifications or tag invalidations received from the primary device will be submitted to the secondary device in order to avoid the failures being seen by the client.

If both the secondary device and the primary device are in an error state, the failed Read or Write that caused the error state will result in a Bad quality update being sent to the client for that item. Any subsequent failures that cannot be handled (due to the devices being in an error state) will be passed along to the client in the form of Bad quality updates or Write Completion failures.

Falling Back to the Primary

If the Operating Mode is set to Switch On Failure and the "Return to Primary Connection ASAP After Failover" setting is enabled, the error state on the primary device will be queried to determine whether it is ready for use each time that a tag is updated (or a write completes on the secondary device). If the primary device is not in an error state, the active device will be switched back to the primary device.

Switchover

The internal switchover process is as follows:

1. Set all client items on the active device to inactive, making note of which were active at the time of the switch.
2. Activate all items on the secondary device that were active on the primary device.
3. Request reads and writes for any items that have these actions pending as the result of a failover.
4. Record the active device time as "Previous" for _PreviousActiveSec .
5. Reset the active device time to 0 seconds.
6. Increment the switch count.

7. Post an Event Log message indicating a switchover has occurred.
8. Update the System Tags.

Monitor Items

The user-defined Monitor Item polls a single item while the active device is in use. This periodic item poll ensures that the inactive device's error state is updated so that it can be switched to when needed. The default setting is 300 seconds; however, when the active device is in an error state, the default setting will be accelerated to a fixed rate of 30 seconds.

When there is no monitor item defined, the first available client item will be used for polling if the secondary device is active. When the secondary is communicating successfully, the default setting is 300 seconds; however, when the secondary device is in an error state, the default setting is 30 seconds.

Note: An item poll will not be issued unless a minimum of one client item exists on the primary device.

Dynamic Changes to Redundancy Settings

Changes made to the redundancy settings will be applied as soon as they are received from the server Configuration. If the secondary device is the active device at the time of the update, the active device time will be reset to zero seconds and a switchover will be recorded. At that time, any items that fail to map to the new secondary device will receive a bad quality update.

Note: If the operating mode is set to Switch On Failure, the primary device will be active when it is not in an error state; otherwise, the secondary device will be active.

Runtime Functions

For more information on a specific aspect of the Media Level Redundancy Plug-In's Runtime functions, select a link from the list below.

[Initialization](#)

[System Tags](#)

[Client Item Management](#)

Initialization

When a project that contains redundant device definitions loads or assigns redundancy settings to a primary device for the first time, the Media Level Redundancy Plug-In will initialize each redundant pair to do the following:

- Create System Tags on the primary device for monitoring redundancy configuration and state.
- Mirror any items to the secondary device that may already exist on the primary device.
- Set the appropriate device (primary or secondary) active, depending on the parameters in the redundancy definition.
- Begin monitoring the primary device for client requests to add/remove items, change an item's active state or data type, or Read/Write an item.

System Tags

Each redundancy definition will have the following Read Only System Tags created in the _System Tag group on the primary device.

Name	Data Type	Description
_ActivePath	VT_BSTR	The fully-qualified path of the active device.
_ActiveSec	VT_UI4	The number of seconds that the active device has been active.
_PreviousActiveSec	VT_UI4	The number of seconds that the previously active device has been active.
_Switchovers	VT_UI4	The number of times the active device has changed.
_OperatingMode	VT_BSTR	The pair's current operating mode.*
_SecondaryPath	VT_BSTR	The current value of the "Secondary Device" parameter.
_FavorPrimary	VT_BOOL	The current value of the "Return to Primary Connection ASAP After Failover" parameter.
_MonitorItem	VT_BSTR	The current value of the "Monitor Item" parameter.
_MonitorIntervalSec	VT_UI4	The current value of the "Poll Monitor Item Every x Seconds" parameter.

*Options include Switch On Failure, Primary Only, and Secondary Only.

Updating System Tags

System Tags will be updated whenever a switchover occurs or the redundancy configuration changes. They are also updated periodically at 500 millisecond intervals. If the redundancy configuration is removed from the primary device, the System Tags that have an active client reference count will receive a bad quality update to indicate that the data is no longer valid.

Note: For more information, refer to [Redundancy Options](#).

Client Item Management

In all operating modes, the primary device will be monitored for add/remove item activity. When this activity is detected, it will be mirrored on the secondary device. Although it may take twice as long to complete item addition, it will greatly reduce the time it takes to perform a switchover (since the necessary tags will already be in place on both devices).

As a side effect of this model, the total tag count will be different for the same client conditions with and without redundancy. For example, a project has one channel and two devices, where each device has one Static Tag defined. If a client maps an item to the tag in Device 1, the total tag count will be reported as one. If Device 2 is then configured as the secondary device for Device 1, the same conditions will result in a total count of two because the tag is mirrored on the secondary device. This increased tag count only affects the server tag count. Clients only see the tag count that is relevant to their primary device.

Note: Having a Monitor Item defined can also affect the tag count.

Adding Client Items

When a client item is added to the primary device, the item mirrored on the secondary device will be generated based on the qualified Item ID received from the client. If the Item ID maps to a Static Tag in the project namespace beneath the primary device, the leading path information will be replaced such that the secondary Item ID will contain the path to the secondary device followed by the remaining path information that identifies the tag. In order for the mapping to succeed, this tag must exist on the secondary device.

Client items added using dynamic addressing are passed as-is to the secondary device for validation and addition. In order for the mapping to succeed, the address must be valid on the secondary device. Any item that cannot be mapped to the secondary device will result in an error message.

Data Type Conversion

The Media Level Redundancy Plug-In will perform data type conversion for any mirrored item that cannot be mapped with the same data type that was used on the primary device. Data type equivalence between the primary and secondary device is determined based on the client data type returned from the tag to which the item is assigned. For example, a type conversion is required when an item mapped to a tag on the primary device uses scaling and the mirrored item is mapped to a tag on the secondary device without scaling, and the scaled data type on the primary device differs from the data type on the secondary device. The same is true when mapping to two scaled tags that have different scaled data types or (two un-scaled tags that have different raw data types).

Note 1: A failed conversion will result in a Bad quality update to the client. A conversion may fail when the range of the two types is different, and the result of the conversion would exceed the range of the target type.

Note 2: In order for the mapping to be successful, the primary and secondary items' data types must be compatible (if not equivalent). For example, if the data type on the primary is an array type and the data type on the secondary is not, then mapping will not be established because a conversion cannot be made between array and non-array types. The same is true when mapping to two arrays that have different dimensions.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Item Validation Error Messages

[' <text>' does not define a tag and is not a valid <driver name> address on <primary path>](#)

[' <text>' is not a valid <driver name> address on <secondary path>](#)

[Monitor item must specify a device tag](#)

[Monitor item specifies a tag that does not exist on the secondary device](#)

[Secondary device '<device path>' cannot be paired with primary '<device path>' because a circular dependency would result](#)

[Specified secondary device '<device path>' cannot be used because the driver type is not supported](#)

[Specified secondary device '<device path>' is the same as the primary](#)

[Specified secondary device '<text>' does not exist](#)

See Also: [Event Log Messages](#)

Event Log Messages

The following Event Log messages may be generated. For more information, refer to the table below.

Event	Type	Message
Change in active device.	Informational	Setting <primary secondary> device configuration active on '<primary device path>'
Removal of redundancy.	Informational	Redundancy disabled on '<primary device path>'
Initialization/Dynamically adding redundancy to a device.	Informational	Redundancy initialized on '<primary device path>' using '<secondary device path>' as the secondary device
Change in secondary device path.	Informational	Secondary device path on '<primary device path>' changed to '<secondary device path>'
Change in settings (other than secondary device path).	Informational	Redundancy configuration updated on '<primary device path>'
Secondary item mapping failure.	Error	Attempt to map redundant item '<qualified id>' to '<secondary device path>' failed

Item Validation Error Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Item Validation Error Messages

[' <text>' does not define a tag and is not a valid <driver name> address on <primary path>](#)

[' <text>' is not a valid <driver name> address on <secondary path>](#)

[Monitor item must specify a device tag](#)

[Monitor item specifies a tag that does not exist on the secondary device](#)

[Secondary device '<device path>' cannot be paired with primary '<device path>' because a circular dependency would result](#)

[Specified secondary device '<device path>' cannot be used because the driver type is not supported](#)

[Specified secondary device '<device path>' is the same as the primary](#)

[Specified secondary device '<text>' does not exist](#)

' <text>' does not define a tag and is not a valid <driver name> address on <primary path>

Reason:

The user-specified text does not map to a Static Tag under the primary device. As such, the device driver rejected it as an invalid device address.

' <text>' is not a valid <driver name> address on <secondary path>

Reason:

The user-specified text is a valid device address on the primary device but not on the secondary device.

Monitor item must specify a device tag

Reason:

A valid tag was specified on the primary device but it does not map to something at the driver level that requires device communications.

Monitor item specifies a tag that does not exist on the secondary device.

Reason:

The user-specified text identifies a tag under the primary device; however, that same tag does not exist on the secondary device.

Secondary device '<device path>' cannot be paired with primary '<device path>' because a circular dependency would result

Reason:

The user-specified text in the Secondary Device parameter specifies an existing device that is not suitable.

Specified secondary device '<device path>' cannot be used because the driver type is not supported

Reason:

The user-specified text in the Secondary Device parameter specifies an existing device, but the channel it belongs to uses a driver that does not meet the requirements of supported use.

Specified secondary device '<device path>' is the same as the primary

Reason:

The user-specified text in the Secondary Device parameter identifies the primary device.

Specified secondary device '<text>' does not exist

Reason:

The user-specified text in the Secondary Device parameter does not specify an actual device.

Index

.

'<text>' does not define a tag and is not a valid <driver name> address on <primary path>... 9

'<text>' is not a valid <driver name> address on <secondary path>..... 9

C

Client Item Management 7

D

Dynamic Changes to Redundancy Settings..... 6

E

Error Descriptions..... 9

Event Log Messages..... 9

F

Failover..... 5

Falling Back to the Primary..... 5

H

Help Contents..... 3

I

Initialization..... 7

Item Validation Error Messages..... 9

M

Monitor item must specify a device tag..... 10

Monitor item specifies a tag that does not exist on the secondary device..... 10

Monitor Items..... 6

O

Overview..... 3

R

Redundancy Options..... 4

Runtime Functions..... 7

S

Specified secondary device '<device path>' cannot be used because the driver type is not ... 10
supported.....

Specified secondary device '<device path>' is the same as the primary..... 10

Specified secondary device '<text>' does not exist 10

Switch On Failure..... 5

Switchover..... 5

System Tags..... 7