Technical Document

Niagara Micros Driver Guide

June 7, 2018

Legal Notice

Tridium, Incorporated

3951 Western Parkway, Suite 350
Richmond, Virginia 23233
U.S.A.

Confidentiality

The information contained in this document is confidential information of Tridium, Inc., a Delaware corporation (Tridium). Such information and the software described herein, is furnished under a license agreement and may be used only in accordance with that agreement.

The information contained in this document is provided solely for use by Tridium employees, licensees, and system owners; and, except as permitted under the below copyright notice, is not to be released to, or reproduced for, anyone else.

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark notice

BACnet and ASHRAE are registered trademarks of American Society of Heating, Refrigerating and Air-Conditioning Engineers. Microsoft, Excel, Internet Explorer, Windows, Windows Vista, Windows Server, and SQL Server are registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Mozilla and Firefox are trademarks of the Mozilla Foundation. Echelon, LON, LonMark, LonTalk, and LonWorks are registered trademarks of Echelon Corporation. Tridium, JACE, Niagara Framework, and Sedona Framework are registered trademarks, and Workbench are trademarks of Tridium Inc. All other product names and services mentioned in this publication that are known to be trademarks, registered trademarks, or service marks are the property of their respective owners.

Copyright and patent notice

This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc.

The product(s) described herein may be covered by one or more U.S. or foreign patents of Tridium.

For an important patent notice, please visit: http://www.honpat.com.

Contents

About this guide

This topic contains important information about the purpose, content, context, and intended audience for this document.

This document is part of the Niagara technical documentation library. Released versions of Niagara software include a complete collection of technical information that is provided in both online help and PDF format. The information in this document is written primarily for Systems Integrators. To make the most of the information in this book, readers should have some training or previous experience with Niagara software, as well as experience working with JACE network controllers.

This guide explains to the Systems Integrator how to configure and use the Micros driver.

Document change log

Updates (changes and additions) to this document are listed below.

June 7, 2018

New release version

- General updates for release

Updated: October 7th, 2016

- Added image showing typical PMS data in the driver
- Added FAQ's for Fidelio/Micros, Guest Data Security and Setup IFC

Updated: October 5th, 2016

- Added Micros Setpoint Select to Palette and Property Reference
- Replace all references to Pre-Check-In with Pre-Arrival

Updated: September 27, 2016

- Numerous typos and corrections
- Added more Property Reference details

Updated: August 31, 2016

- Renamed topic reference ID's for topic_xx's.html
- Deprecated various references to Fidelio
- Added Property Reference Chapter and reorganized Property topics

Updated: April 05, 2016

Add PMS incompatibility.

- Add PMS incompatibility section to Compatibility topic
- Reorganized index references in Compatibility topic

Updated: April 01, 2016

Major restructure of document.

- Micros reference information moved to Appendix
- Getting started and Setting up the station added

Initial release document: February 02, 2016

Related documentation

Several other documents are available for learning how to use the Niagara Micros Driver.

- *Niagara Drivers Guide* explains concepts.
- *NiagaraAX User Guide* explains concepts.

Chapter 1. Getting started

The following topics describe how to get started:

Quick start

The basic steps to configure a station for communication with a Micros Property Management System (PMS) involves setting up Workbench, commissioning the Supervisor or JACE platform and configuring input proxy points. Here is a summary of the steps. They are described in more detail later:

Step 1. Plan the system configuration (Guest room events to report to the station, and resulting values to set).

Step 2. Check and, if necessary, get the PMS Server configured to support communication with the station.

Step 3. Install the Micros driver modules into Workbench.

Step 4. Check and if necessary obtain and install the microsFidelio feature in the Niagara license.

Step 5. Commission the Supervisor or JACE platform.

Step 6. Set up a Micros Network in the station.

Step 7. Configure the IP connection of the Micros Device to the PMS Server.

Step 8. Set up one or more Micros Points in the station.

Step 9. Set up alarms and other components.

Plan system configuration

Both the station and Micros Property Management System (PMS) require configuration. Naming conventions need to be considered and simple system documentation designed.

PMS and station configuration

Implementing the Micros driver involves coordinating the PMS with the station configuration of the Micros driver. Included is any integration with other station control logic. For example, if a Micros proxy point representing a room is set to `true {ok}` when a Guest Check-in event is received by the driver, you may want to reset the room's environmental controls to an occupied state.

Best practices

Before you begin, make a list of the configuration you will need, including:

- You should check for FIAS and PMS (refer to Compatibility).
- Identify which guest rooms are to be monitored by the driver
- Design a naming convention that clearly describes each guest room within the establishment. (Building, Floor, Level, Room, etc.)
- Identify what control strategy is required for Guest Check-in, Guest Check-out and Guest data change events
- Identify if the PMS supports Pre-Arrival (refer to About Pre-Arrival) and, if so, what is the requirement (if any) of the room's environmental control strategy

Compatibility

FIAS compatibility

The driver was developed and tested against the **Fidelio Interface Application Specification** (FIAS) protocol.

Version 2_20w.

CAUTION: The driver *may* work with a number of other versions of the FIAS protocol because the **guest data records** *appear* to have been consistent for many years.

Micros PMS compatibility

The supported Micros Property Management Systems (PMS) are listed below:

- Micros Suite8
- Micros Opera

Third Party PMS compatibility

There are a number of third party PMS systems which *we believe* also implement the FIAS protocol. Some are listed below:

- Amadeus Hospitality
- Avaya HotCom PMS
- Brilliant Galaxy
- Hilton ONQ
- Hotel Concepts
- Hotel Pro
- Newhotel
- Protel
- Silverbyte/Optima
- Tiger TMS iCharge. Refer to Tiger TMS iCharge

CAUTION: The Micros driver implements the FIAS protocol and it *may* operate successfully with other third party PMS systems such as those listed above because many have protocols which *we believe* have been based on FIAS. You should exercise caution and it is strongly recommended that before making any commitment to operate with a third party PMS system you review the Vendor Record Specification Form, compare this with the message types supported by the third party system and test compatibility with an example.

Micros requirements

Requirements include the version of Niagara supported and platform licensing requirements. The PMS system must also be configured.

Systems Integrator requirements

The procedures in this document assume that you:

- Are Niagara certified and experienced at configuring stations.

Platform prerequisites

The Micros driver requires a Niagara 4 platform. The Micros driver only supports an IP connection to the PMS server.

Version of Niagara

The Micros driver requires N4.1 or later.

Licensing requirements

- The **microsFidelio** feature must be present in your Niagara station license.
- Attributes associated with the **microsFidelio** feature are listed below:
- The guest.data or room.rights attributes must be set to true to retrieve these extra details for each point. Typically they would be present as the intention of licensing is to allow disabling where privacy concerns

might exist having guest identifying data within the PMS system.

Attribute	Description
device.limit	This attribute is common to most Niagara features and it defines the maximum number of devices that can be connected to this driver. It is unused in this driver.
guest.data	This attribute enables or disables the collection of extra guest data.
history.limit	This attribute is common to most Niagara features and it defines the maximum number of histories that can be used for this feature. It is unused in this driver.
point.limit	This attribute is common to most Niagara features and it defines the maximum number of points used on this feature.
room.rights	This attribute enables or disables the collection of Room Rights data (Minibar and TV).
schedule.limit	This attribute is common to most Niagara features and it defines the maximum number of schedules used on this feature. It is unused in this driver.

PMS Server configuration

The Micros driver connects via an IFC8 Interface Server. IFC8 is an Interface Server which is a gateway to the PMS (Suite8, Opera, 3rd Party).

- You may require a Micros engineer to visit the site to install and activate the IFC8 Interface Server.
- The minimum IFC8 Version is 7, as earlier versions did not support TCP/IP.

Micros driver modules

Micros modules

The Micros driver comprises the following modules:

- micros-rt.jar
- micros-doc.jar
- micros-wb.jar

Install modules

The latest Micros .jar files may or not be present in the Workbench **Modules** folder. It is important to work with the latest modules.

Prerequisites:

- A version of Workbench that supports the driver must be installed on the PC or laptop computer. For driver requirements, refer toMicros requirements.
- Access to Niagara-Central to retrieve the modules .jar files if you do not already have the latest modules.

Use this procedure to manage the driver modules.

Step 1. Check the Niagara 4-*version* modules folder (where *version* is the version of Niagara 4 you are using).

Step 2. If needed, download the latest module .jar files from Niagara-Central and save them in the Niagara 4-*version* modules folder.

Chapter 2. Setting up the station

The following topics describe how to set up the station:

Add a Micros Network

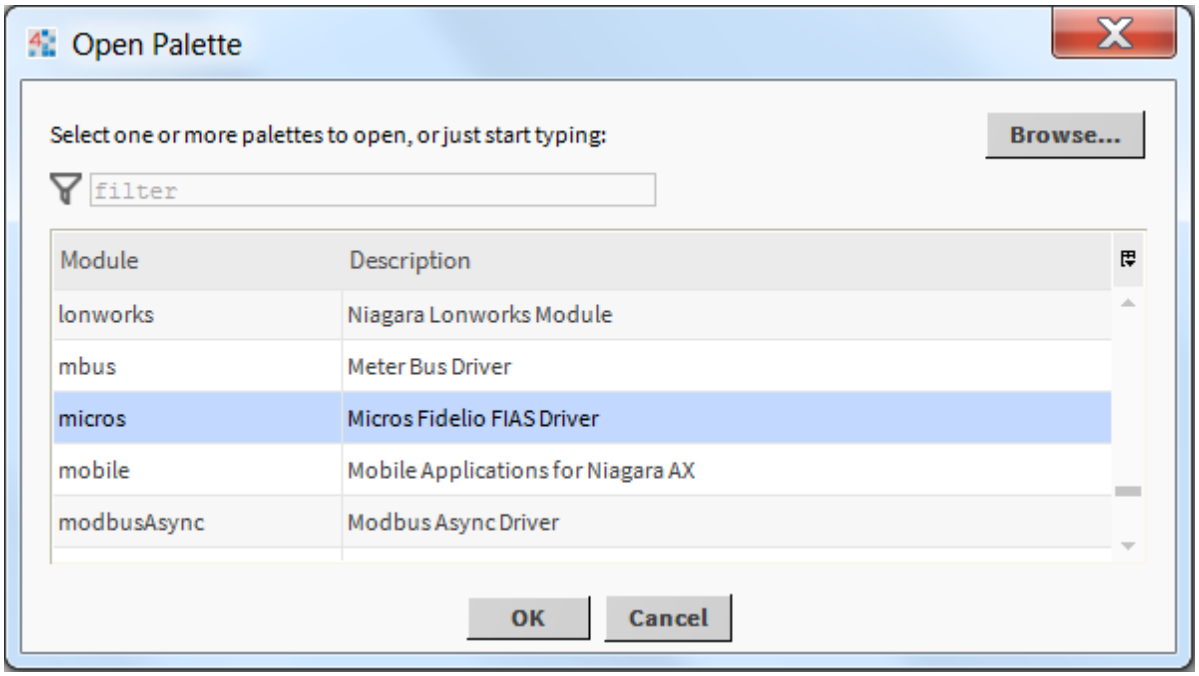
Adding the network is the first step to configure the station.

Prerequisites:

- The driver modules must be available in the Niagara 4-*version*\modules folder (where *version* is the version of Workbench).

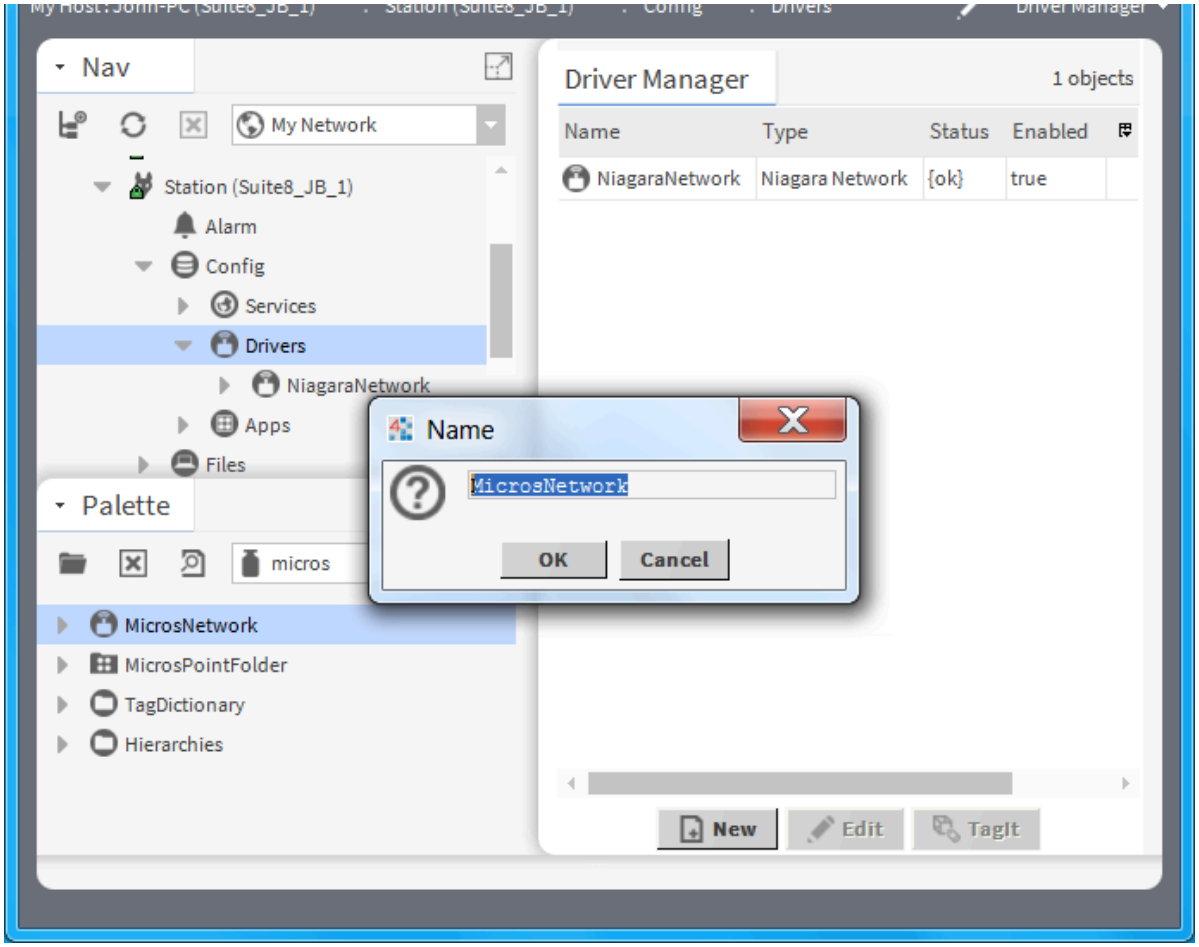
Step 1. In the Nav tree, expand the station and double-click the **Drivers** node.

Step 2. In the **Driver Manager** view, open the driver palette in the side bar.



Step 3. Select the **Module** and click **OK**.

Step 4. Drag or copy a network component from the palette to the **Driver Manager** view pane.



Step 5. Name the network and click **OK**.

Result
You are ready to configure the station using driver features.

Add a Micros Device

The Micros driver supports just one Micros device which is included in the **MicrosNetwork** when added to the **Drivers** node. **Device Discovery** is not supported in this driver.

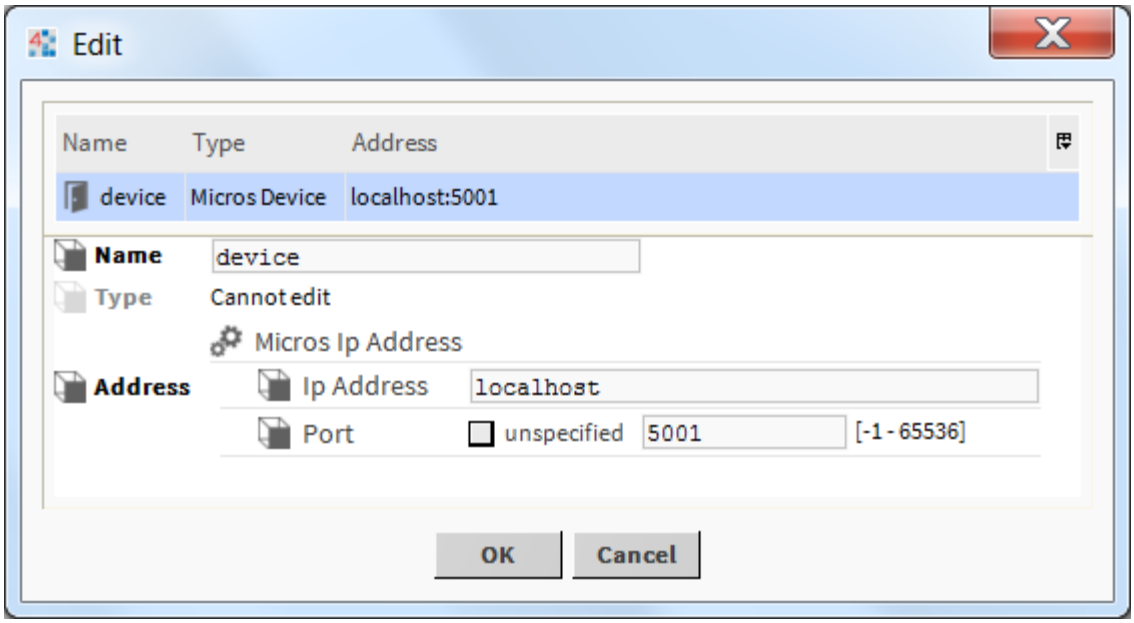
Edit device properties

Editing properties allows you to configure the Micros device.

Prerequisites:

- The Niagara station must be open.

Step 1. In the **Database** pane of the **Micros Device Manager View**, click the device and click **Edit**, or double-click the device.



For a description of each property, refer to Micros driver windows.

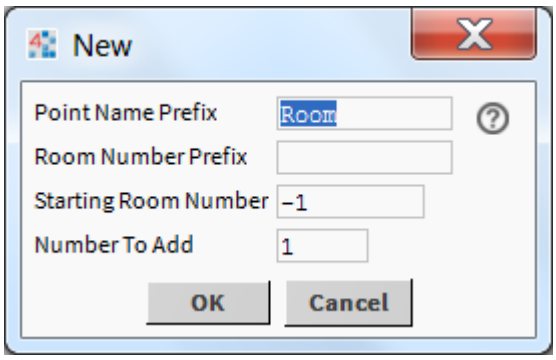
Step 2. Change the properties, confirm the values and click **OK**.
At a minimum you should check or set **Address** to something appropriate for the PMS system connection.

Add Micros Points

Point Discovery is not supported in this driver. Proxy points need to be manually created.

Step 1. Double-click the **Points** folder under the **Drivers** > **MicrosNetwork** > **Device** folder in the Nav tree.

Step 2. Click **New** to add one or more proxy points.



Step 3. Enter a **Point Name Prefix**. This is the text to be used in naming the proxy point.
For example, the guest rooms could be prefixed with the name **Room** `101` or **Room** `102` etc.

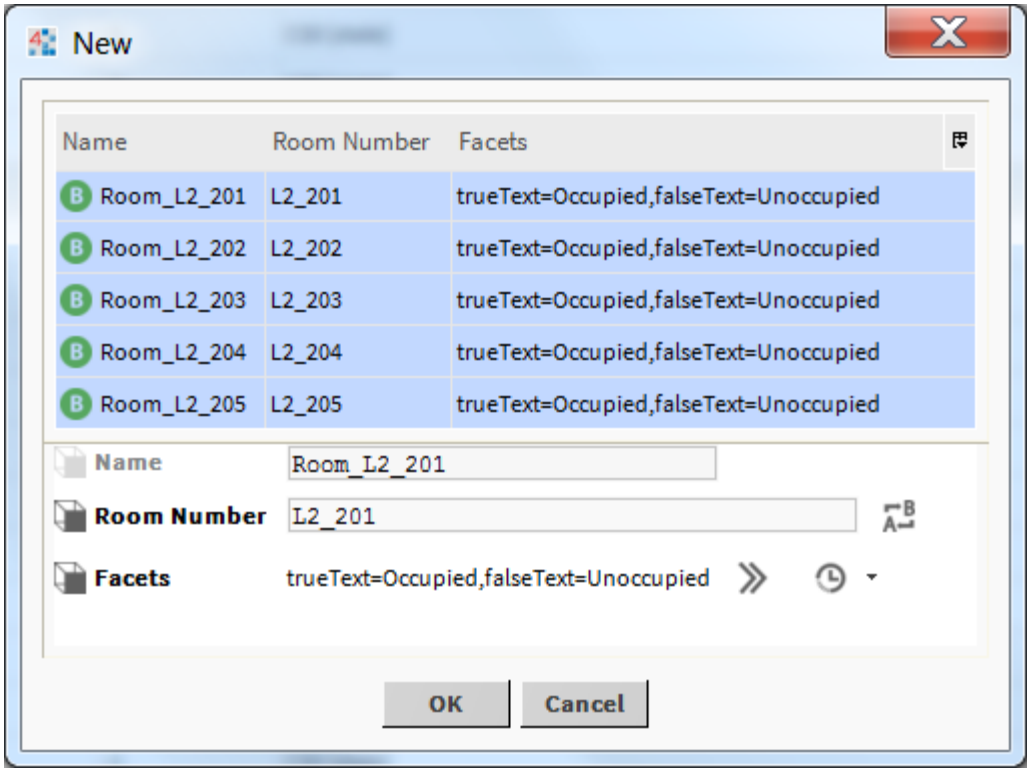
Step 4. Enter a **Room Number Prefix**. This is the leading part of the address within the hotel PMS system.
For example, if the room numbers start with **GROUND** in this set, that will be the prefix used before the number chosen below.

Step 5. Enter the **Starting Room Number**. This is the second part of the hotel room number, where the floor numbering starts.
For example, the first room number you are creating on Level 2 may be `201` .

Step 6. Enter the **Number To Add**. This is how many points to add, incrementing the room number from the starting value above. Click **OK**.

For example, if you have input the following data for the details above: **Point Name Prefix** = Room, **Room Number Prefix** = A, **Starting Room Number** = 10, and **Number To Add** = 5. The result would be 5 new points: RoomA10, RoomA11 through RoomA15.

Step 7. Optionally, change the **Facets** to something more appropriate than true or false. Click **OK**. For example, guest rooms could be Occupied or Unoccupied.



Result
The proxy point(s) are ready to be connected on the wiresheet.

Chapter 3. Micros driver reference

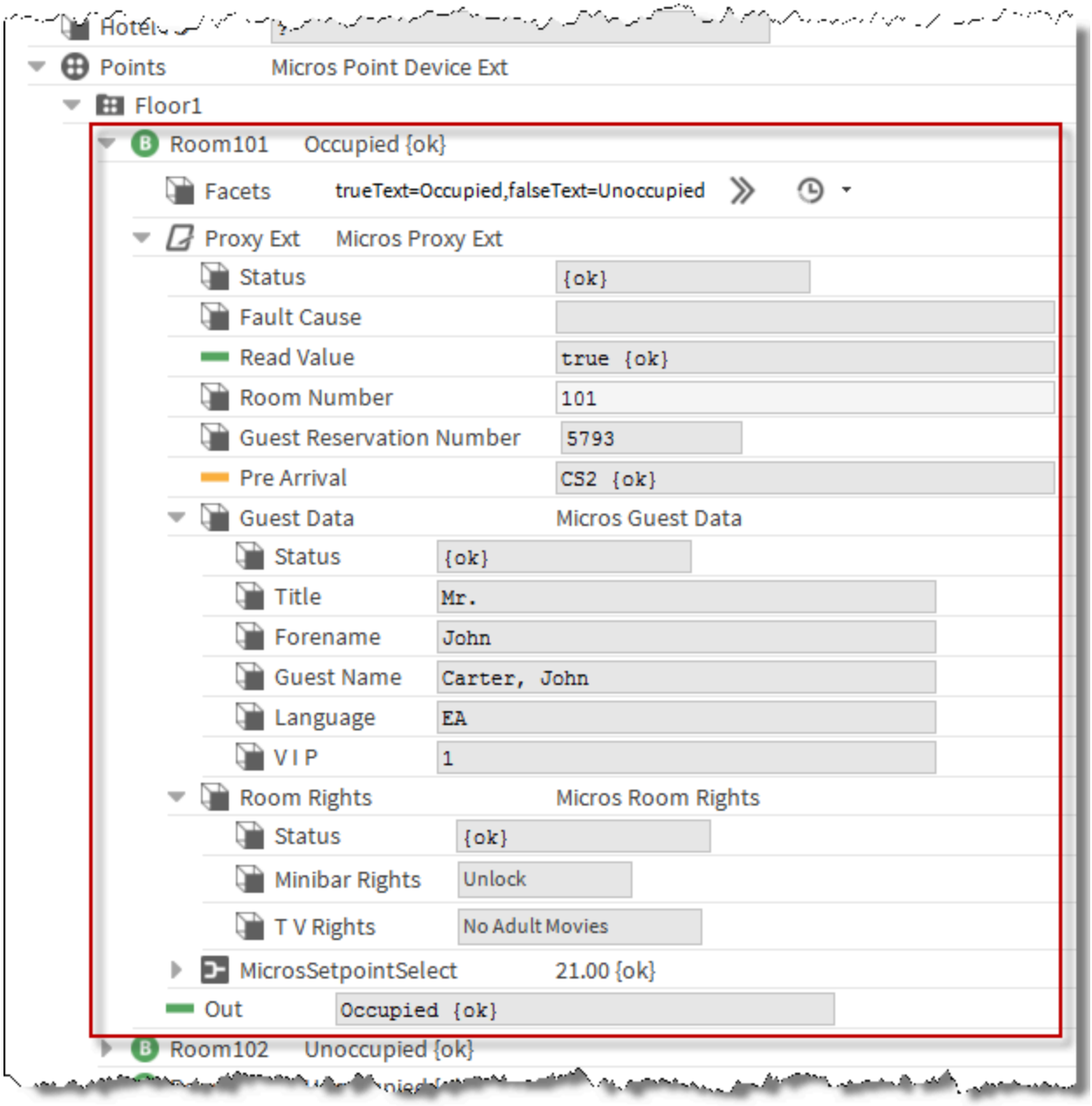
About the Micros driver

This driver connects a Niagara station to a Micros Property Management System (PMS).

The Micros driver integrates a single (PMS) into a Niagara station. Data items in the PMS and the station are exchanged via Micros proxy points under a Micros device. PMS generated Guest Check-in, Guest Check-out, and Guest data change events are sent to a `Boolean` Point to represent the occupancy state of an individual hotel room. This state can then be used to control for example, central plant, room lighting, or the room environment.

The driver uses a familiar Niagara driver hierarchy of a single parent network, a single device, which has one or more proxy points. The room data is **not** polled but instead subscribes to receive PMS event types on startup and thereafter event notifications which use the driver's Micros Network/Link Config.

Here is a typical example of PMS data items in a Micros driver proxy point:



About Micros FIAS

Micros FIAS is a protocol specification which can be used by different kinds of third party systems to exchange data.

Micros FIAS allows a system to define its own specific data records using the list of available record types and fields to create desired functionality to be supported within an interface. The FIAS protocol is widely used in the hospitality industry to connect room equipment and payment systems to the central hotel PMS. It is over 15 years old and is supported by many different vendors and systems.

Other systems using FIAS include:

- Building Management (BMS)
- Energy Management (EMS)
- Call Accounting (CAS)

- Credit Card (EFT)
- Gift Card (SVS)
- Door Locking (KSS)
- Minibar (MBS)
- Point Of Sale (POS)
- Telephone Management (TMS)
- Video Services (VSS)
- Voice Mail (VMS)
- In Room Internet (WWW)

Micros driver summary

Here are some highlights of the Niagara Micros Driver:

- Uses the Micros FIAS Protocol
- Connects via TCP/IP only. (Serial interfaces are not supported)
- Supported only on Niagara 4. (Another version of the driver exists for NiagaraAX)
- It can run on either a Supervisor or JACE platform. (Refer toLarge Installations)

FIAS Protocol Features Implemented

Here is a summary of the features that are implemented in the Micros driver:

Type	Description
Guest Information	Supported Guest records and fields are: (See Vendor Record Specification Form for more details). • Guest Check-in • Guest Check-out • Guest Room Number • Guest Reservation Number • Guest Data • Room Rights • Guest Room Change • Guest Information Change Record Refer to About Guest Information.
Database Sync	Used to update the state of rooms added to the station. Refer to About Database Sync .
Assumed Checkout	This is an implied condition. Refer to About Assumed Checkout .
Pre-Arrival (Energy Management System Scheduler)	Receive notification prior to arrival of the guest to activate in-room devices. Refer to About Pre-Arrival .

Type	Description
VRSF Document	Details the protocol features used by the driver. Refer to Vendor Record Specification Form .

About Guest Information

The following Guest Information is sent from the Micros PMS system to the driver:

- Guest Check-in
- Guest Check-out
- Guest Room Number
- Guest Reservation Number
- Guest Data
- Room Rights
- Guest Room Change
- Guest Information Change Record

Guest Check-in and Guest Check-out

The Micros PMS notifies the driver of Guest Check-in and Guest Check-out when they occur and the driver is always enabled to listen for these two events.

Guest Room and Reservation Number

The Micros PMS notifies the driver of the Guest's Room and Reservation numbers and the driver is always enabled to listen for this data.

Guest Data

Guest Data comprise the Title, Forename, Full name, Language and VIP status of the Guest. By default, the driver will listen for this data but its collection and visibility may be disabled by a license feature (refer to Micros requirements), or disabled by a setting in the Micros Network/Link Config.

Room Rights

Room Rights comprise Minibar Rights and TV Rights. By default, the driver will listen for this data but its collection and visibility may be disabled by a license feature (refer toMicros requirements), or may be disabled by a setting in the Micros Network/Link Config/Room Rights Config.

Guest Room Change

If a guest changes room neither the **old room** nor the **new room** *has* to exist in the Niagara station. If in the hotel, for example, there is a JACE per floor, then the guest will be removed from the **old room** (if it exists it will be set unoccupied), and if the **new room** exists then this will be checked in with the guest details from the PMS. Guest Room change data may be configured by a setting in the Micros Network/Link Config/Room Rights Config

Guest Information Change record

Updates the **Guest number** or **Guest name**, keeping the same room occupancy state. Guest Information Change record data may be configured by a setting in the Micros Network/Link Config/Room Rights Config

About Database Sync

Database Sync is the process of synchronizing the data in the driver with what is currently on site in the PMS system.

In normal operation the Micros PMS notifies the driver of room events such as Guest Check-in and Guest Check-out when they occur. The driver cannot "poll" the PMS but instead subscribes to receive those event types on start-up. It is because of this event driven nature that newly added points have {stale} status until a message is received from the PMS.

To import room occupancy data to a newly configured station, FIAS supports a Database Sync action, which transmits the occupancy state of all the rooms in the PMS to the station.

CAUTION: Performing a Database Sync requires the Hotel's PMS Server to stop communications with *all* its other connected systems while the Database Sync is in progress. Therefore it should be invoked as infrequently as possible. The FIAS *protocol manual* makes the following statement:

"NOTE: A Database-Swap may NOT be requested after every startup. It puts major overhead on communications, especially at larger installations. It should only be requested if data is really not synchronous any more. NEVER request Database Swap requests periodically. Note: Newer Micros systems have been modified in a way, so that unnecessary Swap-Requests will be simply ignored without any notification.— Source — FIAS 2_20w."

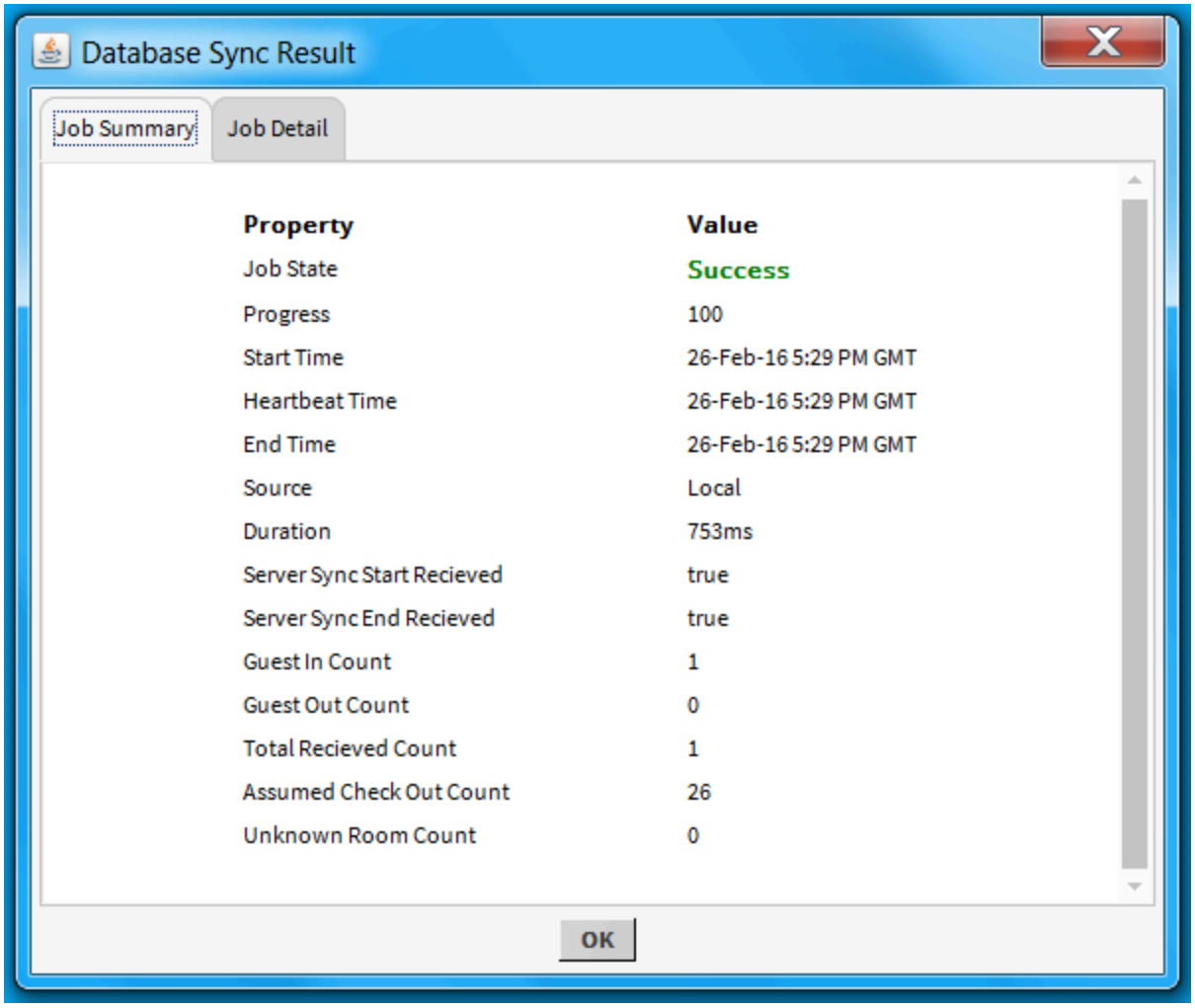
It, therefore, makes sense to request a Database Sync only once all of the room points have been added to the station. The driver will discard update information for rooms that do not have a corresponding `Boolean` point created in the station.

Database Sync can be initiated either:

- By the PMS, for example a front desk operator choosing to push an unsolicited **data update** to the connected station
- By Niagara invoking the **Database Sync** Action on the MicrosNetwork, or by using the **Database Sync** button in the **MicrosPointManager** view

In either of these cases, Niagara handles the event using the **Job Service** which produces various metrics summarizing the exchange. This dialog can be found via the station **Job Service** or by using the **Database Sync Result** button in the **Point Manager** view.

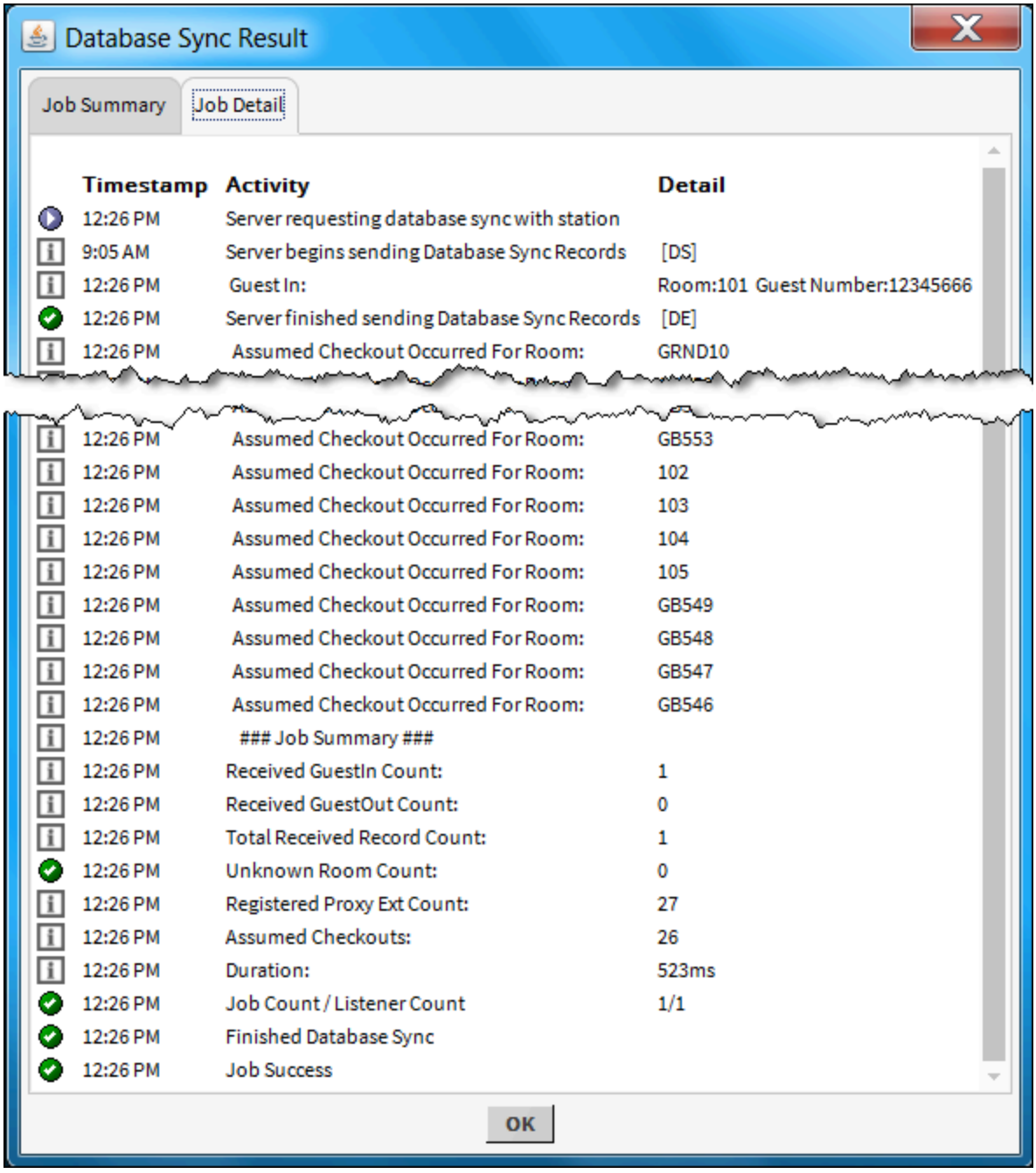
NOTE: The **Source** can be `<Local>` or `<Server>` depending upon who initiated the Database Sync.



NOTE: The PMS system needs to be connected for over 1 minute before it will return records.

About Assumed Checkout

The FIAS protocol suggests that any room which does not receive a Guest Check-in or Guest Check-out message during Database Sync should be considered to be unoccupied. The driver terms this behavior Assumed Checkout . The driver keeps a total count of Assumed Checkouts in the Database Sync **Summary Details** and an example is shown below. More detail on which specific rooms where the occupancy state changed can be found both in the station's **Application Director output** or the **Job Log** view of the Database Sync Job, found by clicking the >> icon in the **Job Service**



In this example, the **Job Count / Listener Count** indicates whether any data, which *should* have been included in the job metrics has *escaped*. The Room data will still maintain an accurate condition but the driver may not have included every change of state in the **Received GuestIn Count** or **Received GuestOut Count** totals. This could be caused by the **Job** data structures not being in place before the PMS started sending data. This situation is only likely to occur in heavily loaded stations.

If you observe that rooms are changing their occupancy state to false {ok} *after* a Database Sync when they were not previously updated by unsolicited data sent normally from the PMS, then you are seeing the Assumed Checkout feature at work.

If you consider this behavior undesirable on a particular site, Assumed Checkout can be disabled by using a property in the Micros Network/Link Config/Database Sync Config. Setting the **Rooms Not Seen During Database Sync** property to preserve occupancy will disable the effects of the Assumed Checkout feature.

About Pre-Arrival

Introduced in version 8.7, **Micros Fidelio Suite 8** now supports a **Pre-Arrival** function (also called **Energy Management System Scheduler**). **Pre-Arrival** is a Room Equipment record which is sent by the PMS to connected BMS systems prior to the expected arrival and room occupancy by a guest.

The use of this function is to allow the BMS to prepare the room's climate control. Unoccupied rooms can be operated in economy mode. With **Pre-Arrival** information there is enough time for climate control to ensure a convenient temperature when a guest enters their room.

CAUTION: If this functionality is a requirement then you should exercise caution and it is strongly recommended that before making any commitment to operate with a third party **PMS** system you check that the connected **PMS** supports **Pre-Arrival**. We are aware that **Micros Fidelio Suite 8** systems prior to version 8.7 and Micros Opera do not support **Pre-Arrival**.

FIAS operation

By default, the **Pre Arrival Status** enumeration in the room point's proxy extension will be set according to any **Pre-Arrival** message received via a FIAS Class of Service (CS) record. Collection and visibility of this data may be disabled by a setting in the Micros Network/Link Config/Pre Arrival Config.

The *FIAS protocol manual* suggests the following: "It is recommended that **BMS** systems support Room Equipment (RE) record with Class Of Service. CS values could e.g. be interpreted as:

- '0' = Aircondition in idle-mode
- '1' = Aircondition 50%
- '2' = Aircondition 75%
- '3' = Aircondition 100%

"These pre-arrival records can only be RoomEquipment (RE) records. Fidelio will send an additional GI record once the guest actually arrives." *Source : FIAS 2_20w*

Niagara operation

Accordingly Niagara allows just four values for **Pre Arrival Status**, the interpretation of which is a local matter. If desired, the value shown by the `Enum` point can be changed in the `lexicon` file using the following keys:

- `re_cs_0 = AC_IDLE`
- `...`
- `re_cs_3 = AC_FULL`

If no **Pre Arrival Status** value is set for a room then the status of the `Enum` point is set to {stale} which indicates the value has not been sent since the last guest event on the room.

A **Micros Setpoint Select Component**, available in the micros palette, when added to the proxy point can reduce the strategy needed to manage setpoint selection with Pre-Arrival signals. Refer to micros-MicrosSetpointSelect component.

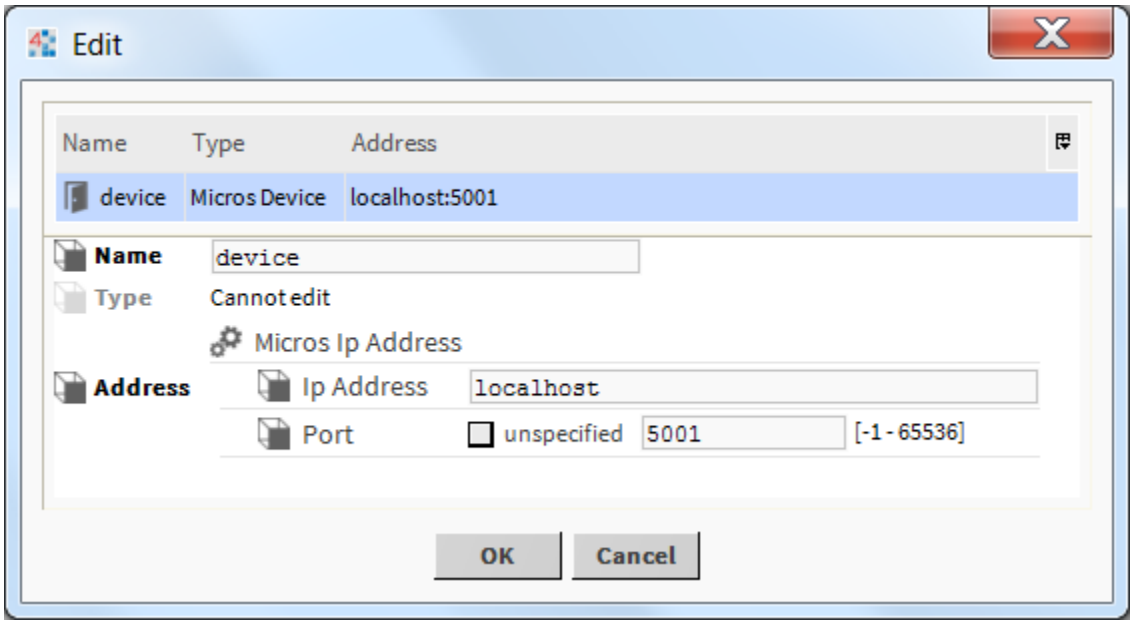
Pre Arrival Status values are not sent during **Database Sync** and the driver sets them to {stale}. We assume **Database Sync** is only invoked when the two systems have inconsistent data.

It should not be considered unusual during normal operation, to observe that a point has normal occupancy data but has a stale **Pre Arrival Status** value. A **Pre Arrival Status** value of {ok} confirms that the value has been sent from the **PMS** server since the last guest movement or **Database Sync** event.

Micros driver windows

The device and point windows appear when editing and adding devices and proxy points.

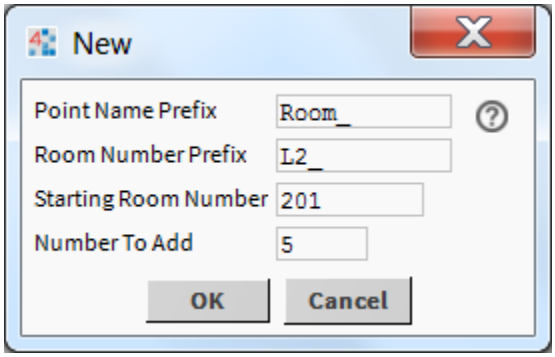
Edit device window

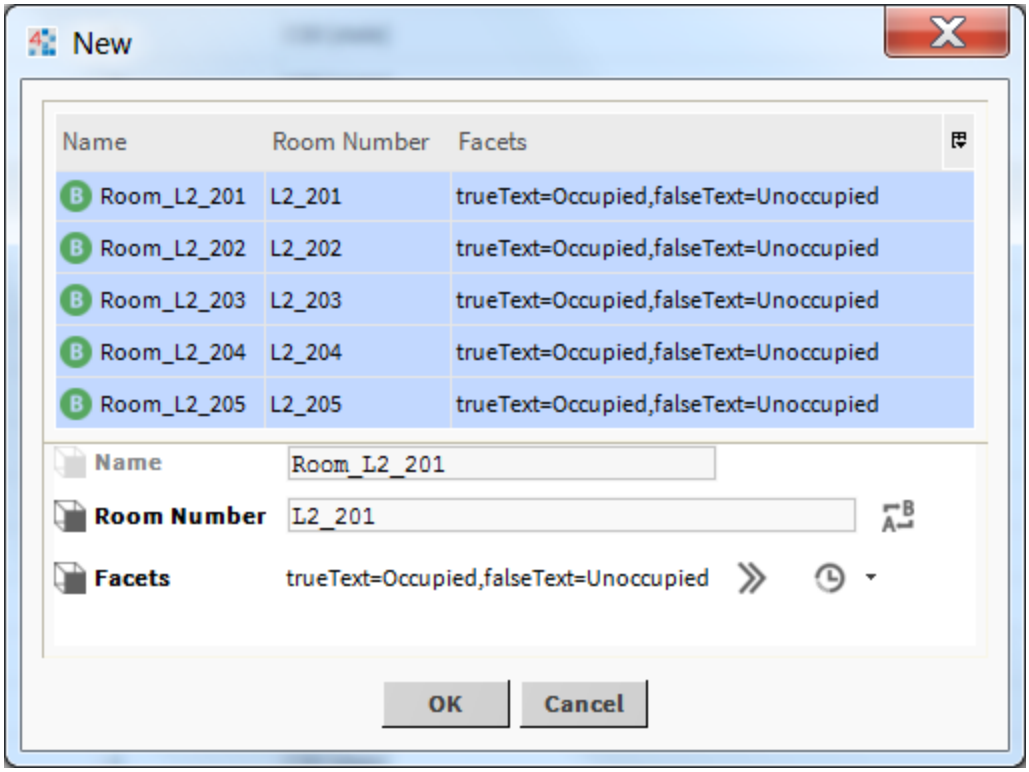


Property	Value	Description
Name	read-only	Identifies the device.
Type	read-only	Indicates the type of point (`Boolean` Point).
Address	IP Address	Defines the address for connecting to the PMS system. (Refer to Micros Device).

Add point window

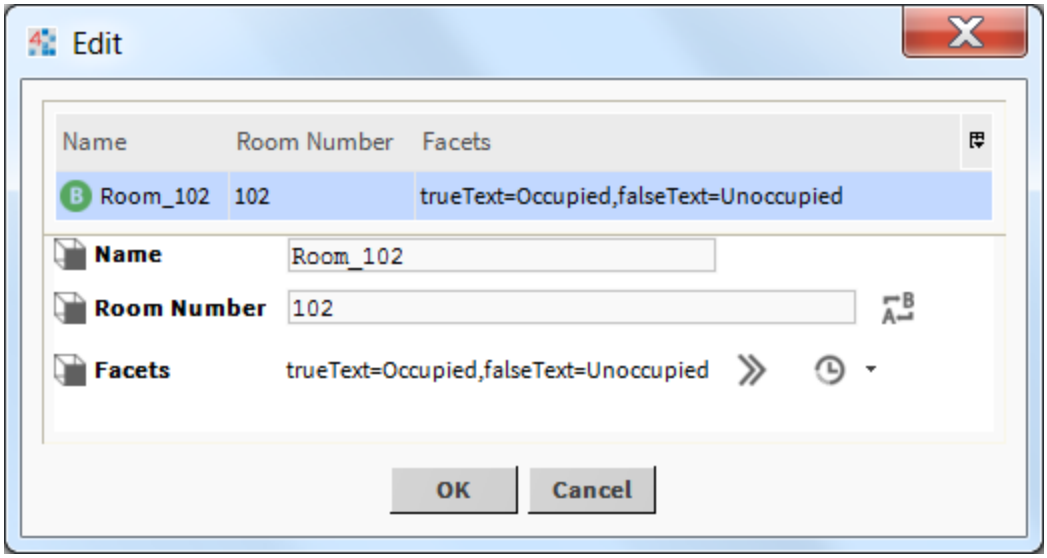
There are two windows associated with adding a **New** point:





Property	Value	Description
Point Name Prefix	text	Describes the proxy point.
Room Number Prefix	string	Defines the leading part of the address within the hotel.
Starting Room Number	integer	Defines the second half of the hotel room number.
Number To Add	integer	Defines the total number of proxy points you are creating.
Name	text	Defines the name of the proxy point.
Room Number	`alphanumeric`	Indicates the room number. **CAUTION:** It is most important to note that the point's **Room Number** must equal the room number (alphanumeric) that the PMS will report events against, i.e the point's **Room Number** must match the room number in the PMS.
Facets	`true` and `false`	Describes the state of the proxy point.

Edit point window



Property	Value	Description
Name	text	Indicates the name of the proxy point.
Room Number	`alphanumeric`	Defines the room number.
Facets	`true` and `false`	Describes the state of the proxy point.

Micros driver components

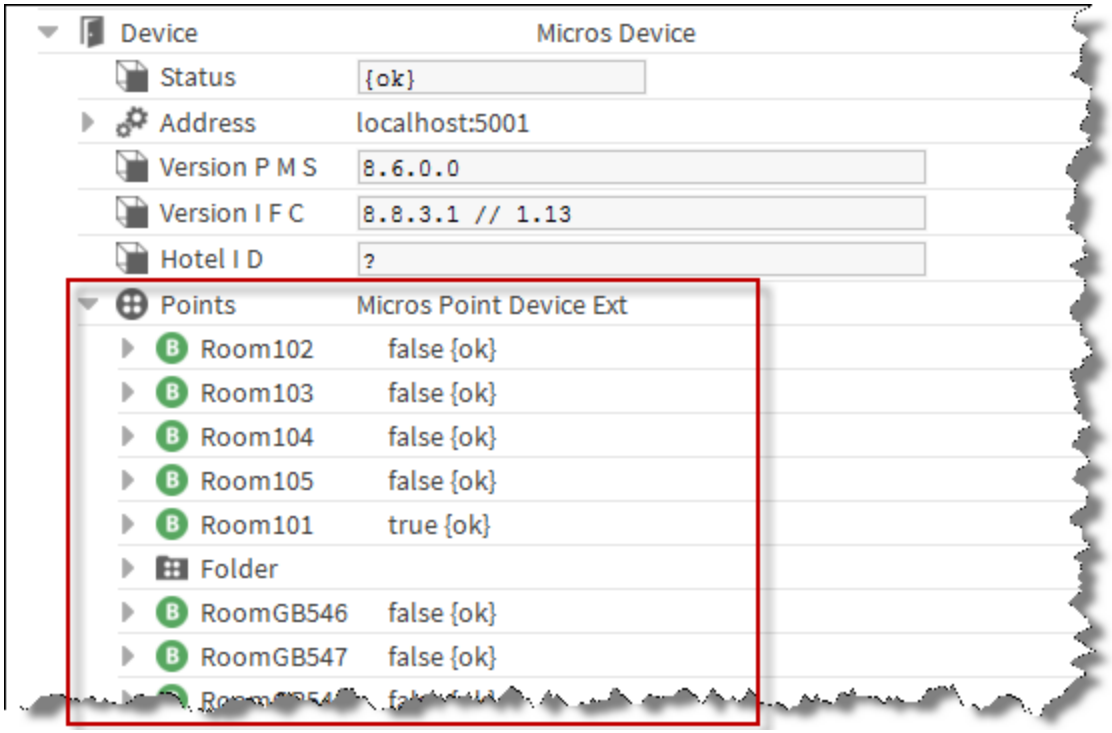
Components include folders and other model building blocks associated with a module. You may drag them to a property or wire sheet from Micros driver components.
The descriptions included in the following topics appear as headings in documentation. They also appear as context-sensitive help topics when accessed by:

- Right-clicking on the object and selecting **Views** > **Guide Help**
- Clicking **Help** > **Guide On Target**

Following is a list of the components in the **micros** module:

Micros Point Device Extension properties

The device's Points extension serves as the top parent container for real-time values originating from the Micros device.



Type	Value	Description
Room102 (example)	`true {ok}` or `false {ok}`	Indicates the occupancy state (`true`) and the status (`{ok}`) of the proxy point (Room)

Micros Point Folder

MicrosPointFolder is the Micros implementation of a folder under a **MicrosDevice** that contains proxy points.

The **MicrosPointFolder** is available in the **micros** palette. The Point Folder allows organization of the room points added to the network, for example by floor or building. This organization may be reflected in a hierarchy as described in Smart Tags. As the protocol subscribes to events from the PMS server, dividing points into separate folders offers no traffic management benefit as would be true for other networks which support polling.

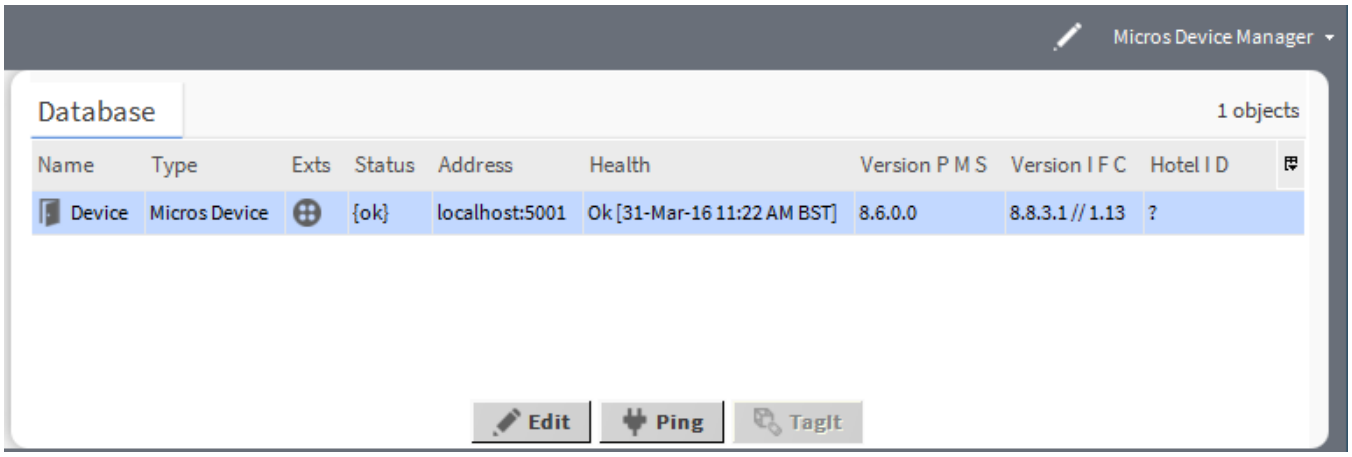
Micros driver component views

Views accompany some Micros driver components.

The following views are provided by the Micros driver:

Micros Device Manager view

The **Micros Device Manager** view displays the Micros devices that have been added to the database.



The database columns indicate the following:

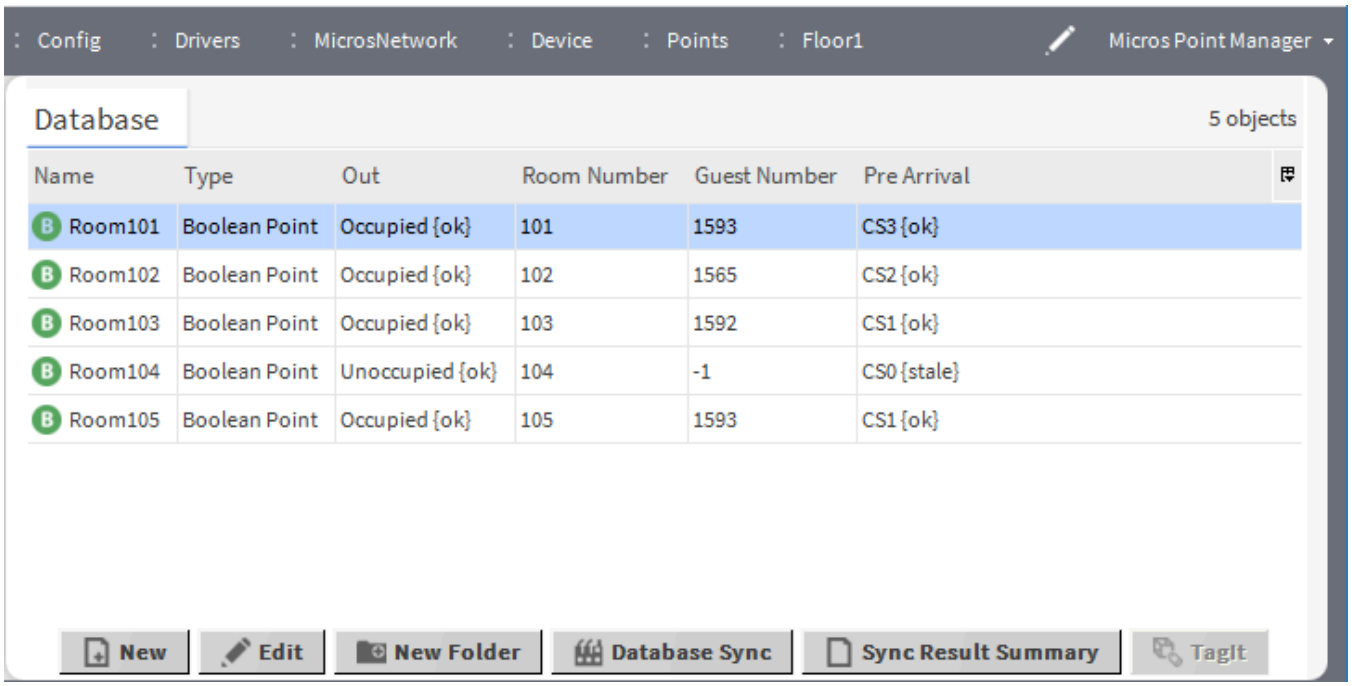
Type	Value	Description
Name	text	The name used when the device was added to the database.
Type	Micros Device	Indicates that this device is a Micros device.
Exts		Provides shortcut access to default manager views for the component's device extensions which are, -in this case, “Points.”
Status	text	Refer to Micros Device .
Address	IP address	Refer to Micros Device .
Health		Refer to Micros Network/Health .
Version PMS		Refer to Micros Device .
Version IFC		Refer to Micros Device .
Hotel ID		Refer to Micros Device .

The buttons initiate the following:

Type	Description
Edit	Initiates the Edit window. Refer to Edit device properties .
Ping	Initiates a ping action to the PMS.
TagIT	This is used to apply direct tags to the device such as location data.

Micros Point Manager view

The **Micros Point Manager** view displays the Micros proxy points that have been added to the database.



The database columns indicate the following:

Type	Value	Description
Name	`(example) Room101`	The name assigned when the point was added to the database.
Type	`Boolean` Point	Indicates the type of point.
Out	`Boolean {status}`	The last received value (out slot) of a proxy point. This value reflects point status and facets.
Room Number	`(example) 101`	The Room number.
Guest Number	`(example) 1593`	The Guest number.
Pre Arrival	`(example) CS3`	The last value of any Pre-Arrival message received via a FIAS Class of Service (CS) record. Refer to About Pre-Arrival .

The buttons initiate the following:

Type	Description	
New		Initiates the New window. Refer to Micros driver windows .

Type	Description	
Edit		Initiates the Edit window. Refer to Micros driver windows .
New Folder		Initiates the New Folder window.
Database Sync		Initiates a Database Sync. Refer to About Database Sync and observe the **Caution**.
Sync Result Summary		Initiates a window of the result of the last Database Sync. Refer to About Database Sync
TagIT		This is used to apply direct tags to the device such as location data.

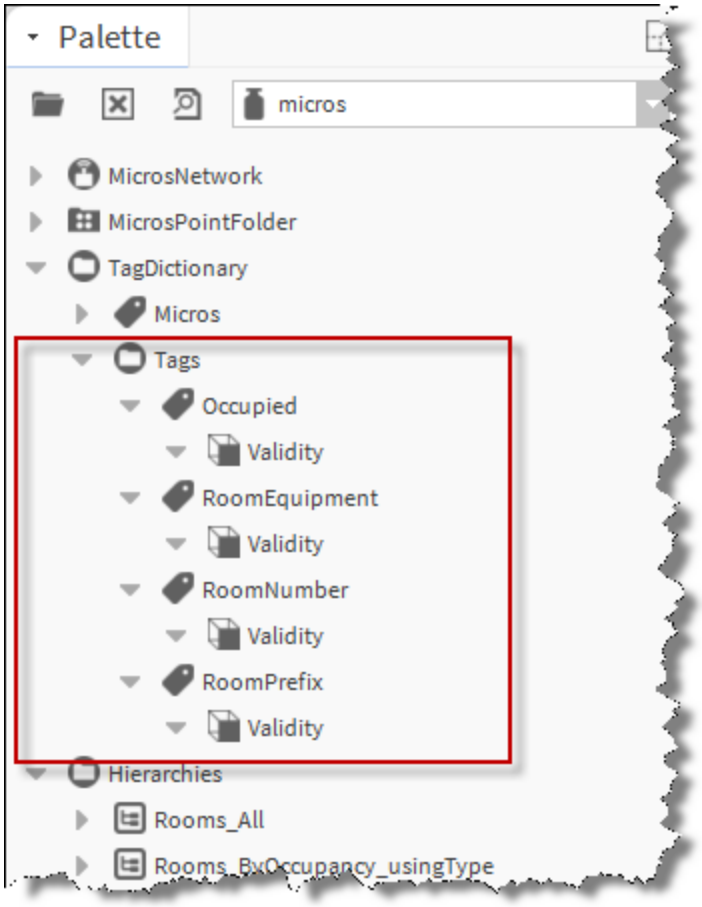
Tags and Hierarchies

A **tag dictionary** containing a list of Tag Rules to assert implied tags for Micros objects is provided. In addition, a number of hierarchy components are available for use. Smart Tags and Hierarchy components are accessible from The micros palette.
The descriptions included in the following topics appear as headings in documentation. They also appear as context-sensitive help topics when accessed by:

- Right-clicking on the object and selecting **Views** > **Guide Help**
- Clicking **Help** > **Guide On Target**

Smart Tags

A number of Smart Tags have been included in the **TagDictionary** in order to leverage the semantic model features of Niagara 4.



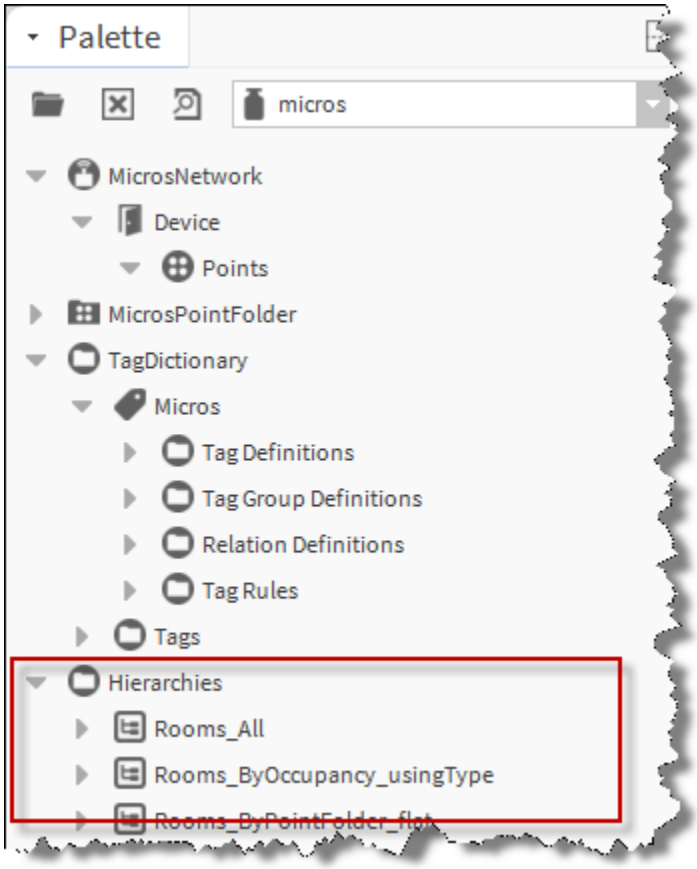
Adding the micros**TagDictionary** to the station's Tag Dictionary Service will automatically apply these tags to any points underneath the station's Micros Device. These could then be used to perform **`neql`** searches by room number or state, or to build a hierarchy for use by the hotel operator. Some examples of possible hierarchies which could be built from these smart tags are included in the **Hierarchies** folder of the micros palette.

Tag Name	Value	Description
Occupied	Boolean	This has a Boolean value reflecting the occupancy state of the Micros proxy point extension.
RoomEquipment	Enumerate	This has an enumerate value reflecting the current `RoomEquipment` setting.
RoomNumber	String	This has a string value reflecting the Micros proxy extensions `Room Number`.
RoomPrefix	text	This presents the first character(s) of a Micros proxy extensions `Room Number` for example: • For a three digit room number (101) it returns the

Tag Name	Value	Description
		first character, eg. 1 • For a four digit room number (1011) it returns the first character, eg. 1 • For a five digit room number (20123) it returns the first 2 characters, eg. 20. • If it starts with a word (for example “GROUND1”) then it returns as much of the "word" as possible

Hierarchies

A number of examples of Hierarchies have been included in the driver.



Hierarchy name	Tags used	Purpose of resulting hierarchy
Rooms_All	`n:type`	All micros points are listed below the root folder
Rooms_ByOccupancy_usingType	`micros:occupied`	Provides navigation of all rooms by `Occupied` or `UnOccupied`.
Rooms_ByPointFolder_flat	`n:type`	Rooms appear grouped by any

Hierarchy name	Tags used	Purpose of resulting hierarchy
		point folders added beneath the device. All are listed from the root node.
Rooms_ByPointFolder_nested	`n:type`	Rooms are grouped by and point folders added beneath the device, however in this case any nesting will be reflected in the hierarchy shown to the user.
Rooms_ByPointFolder_nested_withRootPoints	`n:type` and `n:child`	Rooms are grouped by point folders added beneath the device. In this case any nesting will be reflected in the hierarchy shown to the user.
Rooms_ByRoomName	`micros:roomPrefixPoint`	Groups the rooms by the room number prefix. For example all rooms beginning `1xx` will be assumed to be on the same floor and so automatically appear in a folder named 1. This is possibly the most visually complete point folder based hierarchy.

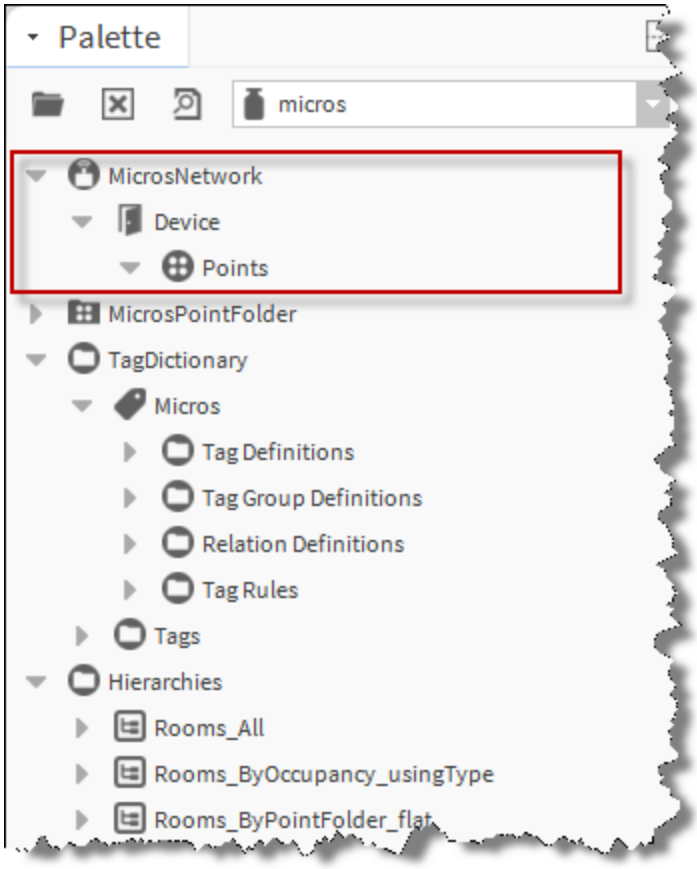
CAUTION: Note that the Micros Tag Dictionary must be added to the Niagara station to support the example hierarchies with the implied tags it introduces.

The micros palette

Following is a list of the components in the **micros** module:

micros-MicrosNetwork Tree

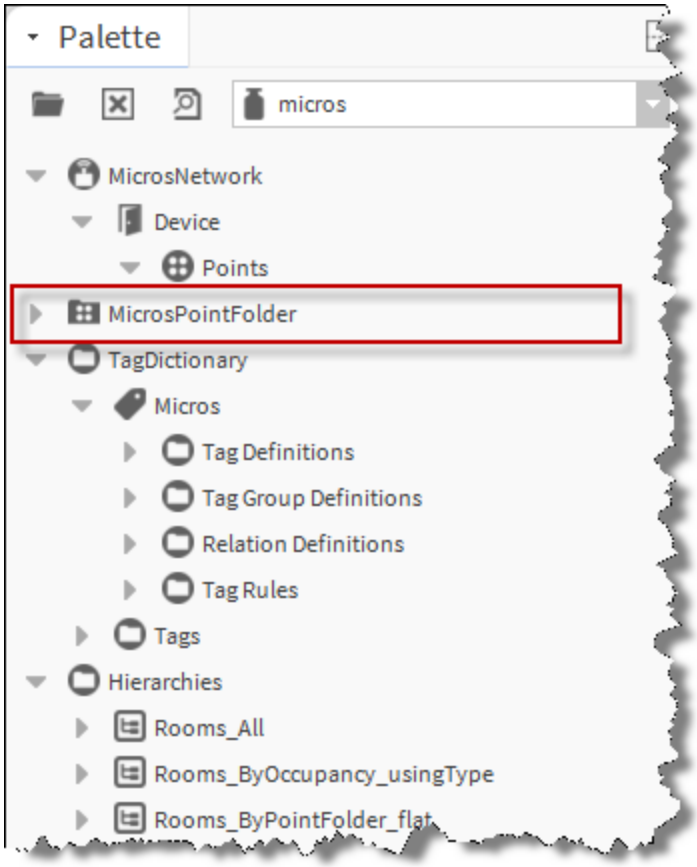
In the palette and station this component appears simply as MicrosNetwork. Its two containers form the primary nodes of the data model structure. You drag it to your **Drivers** container in the Nav tree to begin setting up the model. Refer to Micros Network.



Container	Description
Device	Contains the Micros device. This is the Micros PMS system which the driver is connected to.
Points	Contains the Micros points. These are the Rooms of the Micros PMS system.

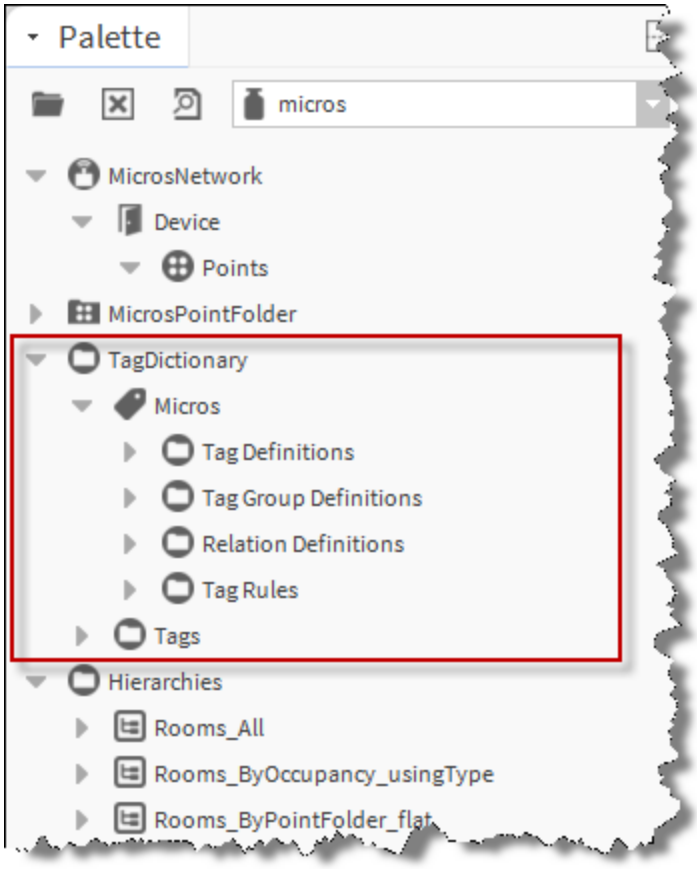
micros-MicrosPointFolder Tree

In the palette and station this component appears simply as MicrosPointFolder. It is a single container which allows organization of the room points added to the network. You drag it to your **Micros Point Device Ext** container in the Nav tree to begin setting up the model.



micros-TagDictionary Tree

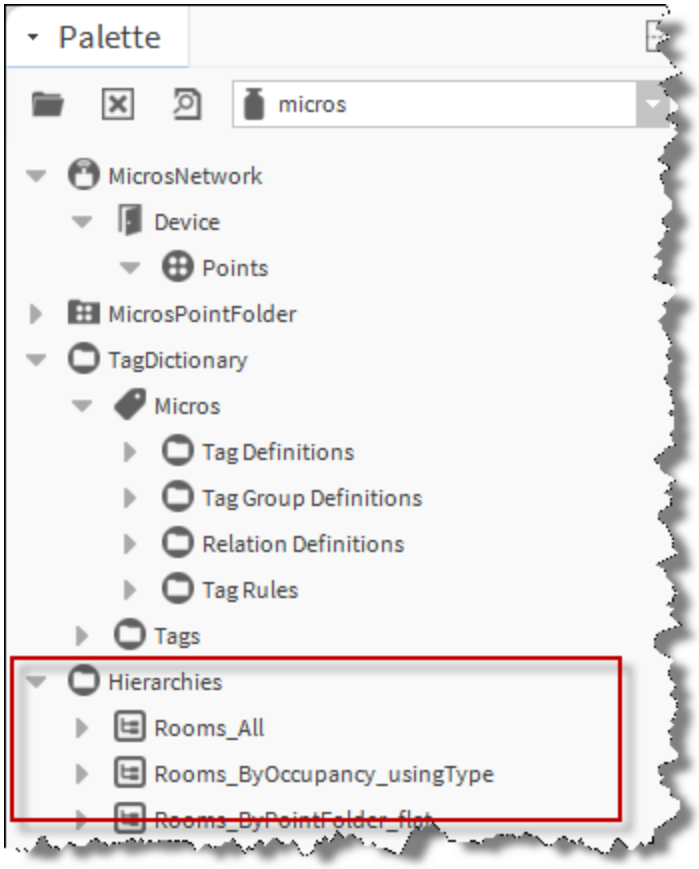
In the palette and station this component appears simply as TagDictionary.



Container	Description
Micros	Contains definitions and rules for Tags, Tag Groups and Relations.
Tags	Contains a number of Micros tags Refer to Smart Tags

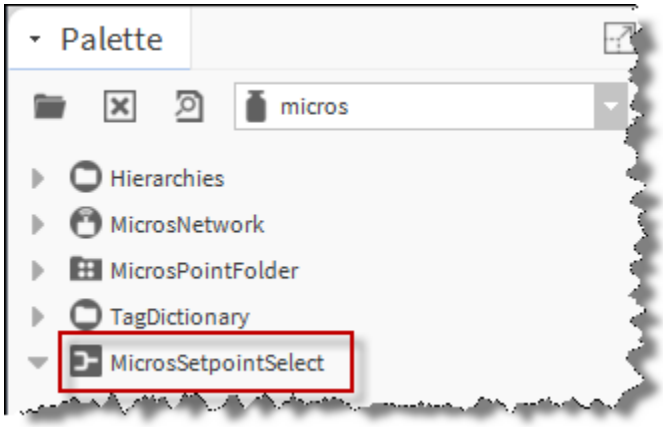
micros-Hierarchies Tree

In the palette and station this component appears simply as Hierarchies. Refer to Hierarchies.



micros-MicrosSetpointSelect component

In the palette and station this component appears simply as MicrosSetpointSelect. It can be added to a Micros Proxy Ext. Refer to micros-MicrosSetpointSelect component.



Chapter 4. Property reference

Micros Network

The **MicrosNetwork** component is the base container for all Micros devices and their child data objects (Micros proxy points).

Properties

To view the MicrosNetwork property sheet, right-click the component and click **Views** > **Property Sheet**.

Property	Value	Description
Status	read-only	Reports the condition of the entity or process at last polling. `{ok}` indicates that the component is licensed and polling successfully. `{down}` indicates that the last check was unsuccessful, perhaps because of an incorrect property, or possibly loss of network connection. `{disabled}` indicates that the **`Enable`** property is set to `false`. `{fault}` indicates another problem. Refer to **`Fault Cause`** for more information.
Enabled	`true` or `false`	Activates and deactivates use of the network.
Fault Cause	read-only	Indicates the reason why a system object (network, device, component, extension, etc.) is not working (in fault). This property is empty unless a fault exists.
Health	text	Health contains historical properties about the relative health of the network in the station, including historical timestamps. Refer to Micros Network/Health
Alarm Source Info	Alarm Source Info	Alarm Source Info contains a set of properties for configuring and routing alarms and are used to populate an alarm if the network does not respond to a monitor ping. Refer to Micros Network/Alarm

Property	Value	Description
		Source Info
Monitor	Ping Monitor	Holds the configuration for the "ping mechanism" used by driver networks. Monitor provides verification of the general health of the network, plus the network's "pingables" (typically, devices) by ensuring that each device is minimally "pinged" at some repeating interval. Refer to Micros Network/Monitor
Link Config		Holds the configuration for the Room Rights, Pre Arrival, Database Sync and enables Guest Data. Refer to Micros Network/Link Config
Link Status	`Active` or `NotActive`	Indicates the condition of the link.
Last link Activation time	`date time`	Displays the last date and time that the link returned to Active
Database Sync Job Ord	ord	The path to the most recently performed database Sync Job in the stations Job Service. Refer to About Database Sync
Device	Micros Device	The Micros Device is the second-tier component for the driver

Micros Network/Health

Health contains historical properties about the relative health of the network in the station, including historical timestamps.

Property	Value	Description
Down	`true` or `false` (default)	Displays the health of the network.
Alarm	`true` or `false` (default)	Reports the component's alarm status. `true` indicates the component has generated an alarm. `false` indicates the component has not generated an alarm.
Last OK Time	`date time`	Displays the last date and time the network health was OK.
Last Fail Time	`date time`	Displays the last date and time

Property	Value	Description
		the network health failed.
Last Fail Cause	text	Displays the reason for the last failure of the network health.

Micros Network/Alarm Source Info

Alarm Source Info contains a set of properties for configuring and routing alarms and are used to populate an alarm if the network does not respond to a monitor ping.

NOTE: For how to format alarm source information on a report, click on the help icon to the right of the field.

Property	Value	Description
Alarm Class	read-only used in Lists, console columns, or %alarmClass% on a report.	Defines alarm routing options and priorities. Typical alarm classes include `High`, `Medium` and `Low`. An alarm class of `Low` might send an email message, while an alarm class of `High` might trigger a text message to the department manager.
Source Name	text	Displays the name in an alarm record that identifies the source of the alarm.
To Fault Text	text	Enters the text to display when the component transitions to a Fault status. When applicable, text entered for **`Fault Algorithm`**, **`High Limit Text`** and/or **`Low Limit Text`** may override this text.
To Offnormal Text	text	Enters the text to display when the component transitions to an Offnormal (alarm) state. When applicable, text entered for **`Fault Algorithm`**, **`High Limit Text`** and/or **`Low Limit Text`** may override this text.
To Normal Text	text	Configures what displays when the component transitions to a normal status. When applicable, text entered for **`Fault Algorithm`**, **`High Limit Text`** and/or **`Low Limit Text`** may override this text.
Hyperlink Ord or Hyperlink	Ord, BQL Query or path	Associates an ORD, BLQ query or path with an alarm state on the component. When an alarm is reported in the console, the

Property	Value	Description
		Hyperlink button activates. Clicking this button links to the location you specify here.
Sound File	ORD	Configures the path to a sound file that plays when the current component is in an alarm state. Use the folder icon to browse to the file. Click the arrow icon to the right of the folder icon to test the path.
Alarm Icon	file path	Defines the location of a graphic file to add to the timestamp column of the alarm table in the Console Recipient view. Use the folder icon to browse to the file that you want to use. Click the arrow icon to the right of the folder icon to test the path that you enter.
Alarm Instructions	**Edit** button	Provides information that may be important or helpful to the user when the system generates an alarm. Click the **Edit** button to open the **Edit** window for working with alarm instructions.
Alarm Meta Data	text	Defines facets for the extension.

Micros Network/Monitor

Monitor holds the configuration for the "ping mechanism" used by driver networks. Monitor provides verification of the general health of the network, plus the network's "pingables" (typically, devices) by ensuring that each device is minimally "pinged" at some repeating interval.

Property	Value	Description
Ping Enabled	`true` (default) or `false`	Controls the monitor ping. • If *true* a ping occurs for the device under the network, as needed • If *false* device status pings do not occur. Moreover, device statuses cannot change from what existed when this property was last true It is recommended you leave Ping Enabled as true in almost all cases.

Property	Value	Description
Ping Frequency	`hours:minutes:seconds`	Specifies the interval between periodic pings of all devices. Typical default value is every 5 minutes (05m 00s), you can adjust differently if needed.
Alarm On Failure	`true` (default) or `false`	Controls the recording of ping failure alarms. • If *true* an alarm is recorded in the station's Alarm database upon each ping-detected device event ("down" or subsequent "up"). • If *false* device "down" and "up" events are not recorded in the station's Alarm database
Startup Alarm Delay	`hours:minutes:seconds`	Specifies how long a station waits at startup before generating a device down or up alarm. Applies only if the **Monitor**'s **`Alarm On Failure`** property is `true`.

Micros Network/Link Config

Link Config contains the settings which define what data is collected from the Micros system.

Property	Value	Description
Msg Retry Count	`1-5` Defaults to 1	The number of times a message will be resent if a response is not received.
Msg Response Timeout	`hh mm ss (1 sec – 10 minutes)` Defaults to 10 secs	The delay before a message (eg ping or Database Sync request) is considered timed-out.
Inhibit Link End Next Shutdown	`true` or `false` (default)	Inhibit the **Link End** routine, sent on shutdown so the server should buffer guest events and resend when the station reconnects. This is useful during a temporary reboot. Will reset to false at the next start.
Enable Guest Data	`true` (default) or `false`	Set this property to `false` if the collection of this data is not required.
Room Rights Config		This defines the configuration for how the Micros driver handles Room Rights. Refer to Micros Network/ Link Config/Room Rights Config .

Property	Value	Description
Pre Arrival Config		This defines the configuration for how the Micros driver handles Pre Arrival. Refer to Micros Network/ Link Config/Pre Arrival Config .
Database Sync Config		This defines the configuration for how the Micros driver handles Database Sync. Refer to Micros Network/Link Config/Database Sync Config .

Micros Network/Link Config/Room Rights Config

This defines the configuration for how the Micros driver handles Room Rights.

Property	Value	Description
Enable Room Rights Data	`true` (default) or `false`	Set this property to `false` if the collection of this data is not required.
On Guest Departure	`reset to default` (default) or `retain in room`	On Guest Departure.
On Guest Change	`transfer with guest` (default) or `retain in room`	On Guest Change.

Micros Network/Link Config/Pre Arrival Config

This defines the configuration for how the Micros driver handles Pre Arrival.

Property	Value	Description
Enable Pre Arrival	`true` (default) or `false`	Set this property to `false` if the collection of this data is not required.
On Guest Departure	`reset to default` (default) or `retain in room`	On Guest Departure.
On Guest Change	`transfer with guest` (default) or `retain in room`	On Guest Change.

Micros Network/Link Config/Database Sync Config

This defines the configuration for how the Micros driver handles Database Sync.

Property	Value	Description
Rooms Not Seen During Database Sync	`set unoccupied` (default) or `preserve occupancy`	Rooms Not Seen During Database Sync. See About Assumed Checkout .
Database Sync Timeout	`hh mm ss (1 s – 6 mins)` Defaults to 30 s.	Set the time after which, if no room data is received, the Database Sync job will fail.

Micros Device

The Micros Device is the second-tier component for the driver

Property	Value	Description
Status	read-only	Reports the condition of the entity or process at last polling. `{ok}` indicates that the component is licensed and polling successfully. `{down}` indicates that the last check was unsuccessful, perhaps because of an incorrect property, or possibly loss of network connection. `{disabled}` indicates that the **`Enable`** property is set to `false`. `{fault}` indicates another problem. Refer to **`Fault Cause`** for more information.
Address	IP address	The IP address of the Micros device. Refer to Micros Device/Address .
Version P M S	`?` (default) or `n.n.n.n`	Upon connection to the server the device object may be populated with version information. For example 8.6.0.0. Seeing data in this property is a positive indication that communication has been established with the PMS. It is reset to "?" at station startup.
Version I F C	`?` (default) or `n.n.n.n // n.n`	Upon connection to the server the device object may be populated with version information. For example 8.8.3.1 // 1.13. Seeing data in this property is a positive indication that communication has been established with the PMS. It is reset to "?" at station startup.
Hotel ID	`?` (default) or `<Hotel ID>`	Upon connection to the server the device object may be populated with hotel ID information. For example HH123. Seeing data in this property is a positive indication that communication has been established with the PMS. It is reset to "?" at station startup.
Points	Micros Point Device Ext	The device's Points extension serves as the top parent container for real-

Property	Value	Description
		time values originating from the Micros device. Refer to Micros Point Device Extension properties .

Micros Device/Address

Micros Device Address contains a set of properties for configuring the Micros Device.

Property	Value	Description
Address	IP address	The IP address of the source or destination device. If a "hostname" is used in place of the `IP Address`, the JACE may require a DNS server to be configured in the **TCP/IP Settings** of the platform.
Port	`nnnn`	Where `nnnn` indicates the Port number. The typical port range for the PMS is 5000 to 6000. The **unspecified** check box is **NOT** used in this driver.

Micros Proxy Ext

The Proxy Extension contains data sent from the Micros device to the station. It is not able to poll but instead receives updates via the **Event Listener** when they are sent from the PMS.

Property	Value	Description
Status	text	Read-only field. Indicates the condition of the component at the last event. • {ok} indicates that the component is licensed and data is being received. • {stale} indicates that data has not been received for this room since the point was added or since the last Database Sync occurrence. • {down} indicates that the component cannot receive data. • {disabled} indicates that the **`Enable`** property is set to `false`. • {fault} indicates another problem.
Fault Cause	read-only	Indicates the reason why a system object (network, device, component, extension, etc.) is not working (in fault). This

Property	Value	Description
		property is empty unless a fault exists.
Read Value	boolean	Read-only field. Indicates the last value received and includes the "status Flag".
Room Number	`(example) 101`	Indicates the address of the room in the PMS system.
Guest Reservation Number	`(example) 1577`	Read-only field. Indicates the last value received of the "Guest Reservation Number". Defaults to -1 if no value has been received.
Pre Arrival	`(example) CS3 {status}`	Read-only field. The last value of any Pre-Arrival message received via a FIAS Class of Service (CS) record. See About Pre-Arrival . If the Micros Network/Link Config has been set `false` then the value of this property will indicate {disabled}.
Guest Data	Micros Guest Data	This is a Read-only container. Contains the guest data received from the Micros PMS. See Micros Proxy Ext/Guest Data . The visibility of this container is controlled by a setting in the Micros Network/Link Config.
Room Rights	Micros Room Rights	Read-only container. Contains the guest Room Rights data received from the Micros PMS. See Micros Proxy Ext/Room Rights . The visibility of this container is controlled by a setting in the Micros Network/Link Config.
Out	`boolean {status}`	Read-only field. This value indicates the last value received of the point status and facets.

Micros Proxy Ext/Guest Data

Guest Data properties specifically contain the guest data received from the Micros PMS.

The visibility of these properties is controlled by a setting in the Micros Network/Link Config.

Property	Value	Description
Status	text	Read-only field. Indicates the condition of the component at the last event. • {ok} indicates that the component is licensed and data is being received. • {stale} indicates that data has not been received for this room since the point was added or since the last Database Sync occurrence. • {down} indicates that the component cannot receive data. • {disabled} indicates that the **`Enable`** property is set to `false`. • {fault} indicates another problem.
Title	text	Read-only field. Indicates the 'Title' of the Room's Guest. (e.g. "Mr").
Forename	text	Read-only field. Indicates the 'Forename' of the Room's Guest. (e.g. "John").
Guest Name	text	Read-only field. Indicates the full 'Guest Name' of the Room's Guest. (e.g. "Smith, John").
Language	text	Read-only field. Indicates the **Guest Language** of the Room's Guest. For example, Guest Language: EA English American, GE German, FR French, IT Italian, JA Japanese, SP Spanish.
VIP	text	Read-only field.

Micros Proxy Ext/Room Rights

Room Rights properties specifically contain the guest Room Rights data received from the Micros PMS.

The visibility of these properties is controlled by a setting in the Micros Network/Link Config.

Property	Value	Description
Status	text	Read-only field. Indicates the condition of the component at the last event. • {ok} indicates that the

Property	Value	Description
		component is licensed and data is being received. • {down} indicates that the component cannot receive data. • {disabled} indicates that the **`Enable`** property is set to `false`. • {fault} indicates another problem.
Minibar Rights	text	Read-only field. Indicates the condition of the component at the last event. • Unlock indicates that the guest room Minibar rights is Unlocked. • Normal indicates that the guest room Minibar rights is Normal Vending (default). • Lock indicates that the guest room Minibar rights is Locked.
TV Rights	text	Read-only field. Indicates the condition of the component at the last event. • Unlimited Pay Channels indicates that the guest room Pay TV rights is Unlimited (default). • No Pay Movies indicates that the guest room Pay TV rights is No Pay Movies. • No Adult Movies indicates that the guest room Pay TV rights is No Adult Movies. • No TV indicates that the guest room Pay TV rights is No TV rights.

Micros Proxy Ext/Micros Setpoint Select

The **Micros Setpoint Select** properties can be added to the **Micros Proxy Ext** to try and reduce the strategy needed to manage setpoint selection with the **Pre Arrival** signals.

The **Micros Setpoint Select** component can be found in the Palette. See micros-MicrosSetpointSelect component.

Property	Value	Description
Out	`value (facets) {status}`	Current active temperature for the guest room (read only).
Facets	`units`, `precision`, `min` and `max` values	Sets the units applied to the `Out` property e.g. deg C
Propagate Flags	Defaults to `null`	Defines which input status `Factory` will be propagated from input to output.
In Occupancy Select	`true {status}` or `false {status}`	Is the room currently occupied or not? (read only). Automatically set by the parent Proxy Ext.
In Pre Arrival Select	`CS0 {status}` or `CS1 {status}` or `CS2 {status}` or `CS3 {status}`.	The input which selects one of the four Pre-Arrival values (CS0, CS1, CS2 and CS3), or Guest Setpoint when {stale} linked to room equipment (read only). Automatically set by the parent Proxy Ext.
In Occupancy Setpoint	`Numeric {status}`. Defaults to 21.00	Set the In Occupancy Setpoint.
In Pre Arrival Setpoint	`Numeric {status}`. Defaults to 18.00	Set the Pre Arrival Setpoint
Fallback Setpoint	`Numeric {status}`. Defaults to 5.00	The Setpoint to use if the calculation fails.
In Pre Arrival Scale Cs0	`Numeric % (0-100)`. Defaults to 25	The Setpoint for Pre Arrival CS0
In Pre Arrival Scale Cs1	`Numeric % (0-100)`. Defaults to 50	The Setpoint for Pre Arrival CS1
In Pre Arrival Scale Cs2	`Numeric % (0-100)`. Defaults to 75	The Setpoint for Pre Arrival CS2
In Pre Arrival Scale Cs3	`Numeric % (0-100)`. Defaults to 100	The Setpoint for Pre Arrival CS3

Chapter 5. Vendor Record Specification Form

The Vendor Record Specification Form (VRSF) is a statement made by Tridium, of the conformance to the FIAS protocol.

The VRSF comprises a number of sections which are detailed here:

Vendor Details

Detail	Description
Product Name	Niagara
Vendor Name	Tridium Europe Ltd.
Address	1 The Grainstore Brooks Green Road Coolham W. Sussex RH13 8GR
Contact Person	Product Manager
Email	supportemea@tridium.com
Tel / Fax Number	+44 (0) 1403 740290
Product Type	IOT & Building Integration Framework
Vendor Software Version/Release Number	4
Target Beta Hotel	TBC
FIAS Version	FIAS 2_20w
Low Level Protocol	TCP/IP

Field Types

The Tridium interface will use only the link, guest and room data field types through a TCP/IP connection.

Communication and link control

1.1 Link Description

->_LDlDAstrong>150926l

->_LDlDAstrong>150926lTIstrong>120215lV#strong>4

Niagara is an Energy Management Interface

1.2 Link Record

-> _LR|RIstrong>GI|FLstrong>RNG#<u class="ph u">GTGFGNGLGVu>SF|_

-> _LR|RIstrong>GI|FLstrong>RNGSG#SF|_

-> _LR|RIstrong>GC|FLstrong>RNGSG#<u class="ph u">GTGFGNGLGVu>R

-> _LR|RIstrong>RE|FLstrong>RNCS<u class="ph u">MRTVu>|_

Niagara requests guest and room equipment data

Guest data (GT,GF,GN,GL,GV) and room access right (MR,TV) fields are underlined in the above text. These may be *disabled* by license if the site requires that the Niagara interface be prevented from accessing this kind of data.

1.3 Link Configuration

<- <strong class="ph b">_LC|DAstrong>150926|<strong class="ph b">TIstrong>124811|<strong class="ph b">A1strong>

Micros replies with its version details

1.4 Link Start & Link Active

<-> _LS|DAstrong>150926|TIstrong>110241|_

<-> _LA|DAstrong>150926|TIstrong>120215|_

1.5 Link End

<-> _LE|DAstrong>150926|TIstrong>131105|_

Database Synchronization

2.1 Database Resync request

-> _DR|DAstrong>150818|TIstrong>160528|_

2.2 Database Resync Start

<- <strong class="ph b">_DS|DAstrong>150818|<strong class="ph b">TIstrong>160528|_

2.3 Database Resync End

<- <strong class="ph b">_DE|DAstrong>150818|<strong class="ph b">TIstrong>160528|_

Guest Data

3.1. Guest Check-In

<- <strong class="ph b">_GI|RNstrong>102|<strong class="ph b">G#strong>12345666|<u class="ph u"><strong class="p

Sent from Micros to vendor upon guest Check-in, or during DB Sync (add SF). Underlined fields can be optionally disabled if not desired.

3.2. GC – Guest Information Change record

<- <strong class="ph b">_GC|RNstrong>101|<strong class="ph b">G#strong>654321|<u class="ph u"><strong class="ph

Sent from Micros to vendor when guest information changes. Underlined fields can be optionally disabled if not desired.

3.3. GC - Room Change record

<- <strong class="ph b">_GC|RNstrong>101|<strong class="ph b">G#strong>12345666|<u class="ph u"><strong class="

Sent from Micros to vendor when a guest changes room. Underlined fields can be optionally disabled if not desired.

3.4. Guest Check-Out

<- <strong class="ph b">_GO|RNstrong>102|<strong class="ph b">GSstrong>N|<strong class="ph b">G#strong>123456

<- <strong class="ph b">_GO|RNstrong>102|<strong class="ph b">GSstrong>N|<strong class="ph b">G#strong>123456

Sent from Micros to vendor upon guest Check-out, or during DB Sync, (add SF).

Room Data

4. Room Data

<- <strong class="ph b">_RE|RNstrong>102|<strong class="ph b">CSstrong>3<u class="ph u"><strong class="ph b">M

Sent from Micros to vendor to signal pre-arrival. Underlined fields can be optionally disabled if not desired.

Chapter 6. Micros driver FAQ's

NiagaraAX to Niagara 4 Migration

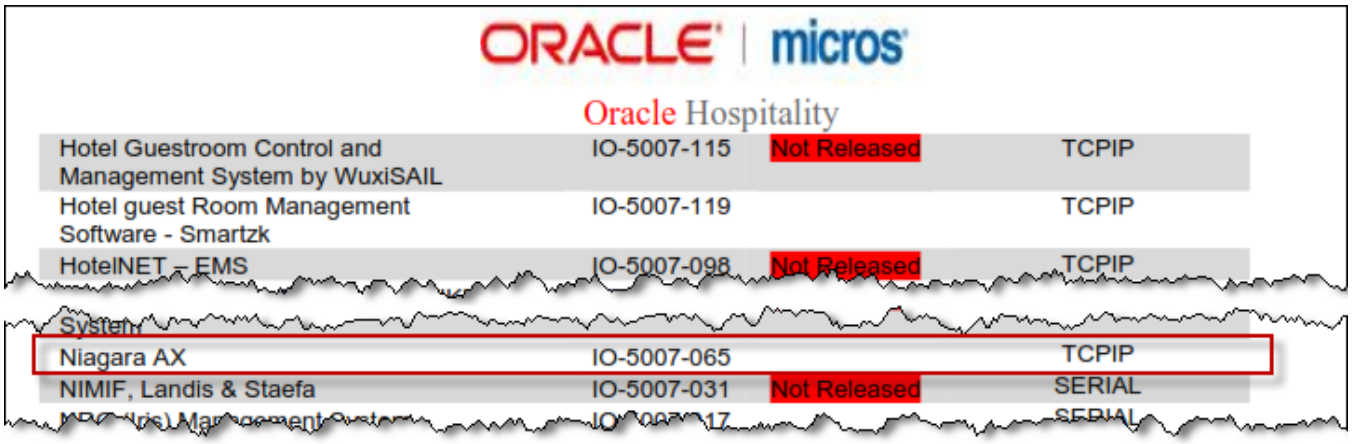
Can an older NiagaraAX station using Micros be migrated to this Niagara 4 driver?

A migrator for all driver settings has not been provided, however the most labor intensive items such as room points, their addresses and onward links will migrate from NiagaraAX to Niagara 4. Properties which contain connection details such as IP address and the port number for the hotel server will need to manually added after migration. Backward conversion from Niagara 4 to NiagaraAX is not possible.

Driver Certification

Is the Micros Driver certified?

The NiagaraAX Micros driver is listed in the `<Oracle>` **Approved Interfaces for Oracle Hospitality OPERA and Suite8 Products** document with Part Number IO-5007–065.



The driver has been updated, thereby enabling partners to use the driver with the latest Niagara 4 software. However, the driver has not been re-certified since the update changes were essentially internal to the Niagara Framework.

FIAS Security

Does the FIAS protocol have any form of security?

No. The FIAS protocol is intended for use only over a local area network. Having no authentication it relies on the security of the network. This should not be run on the same physical network that hotel guests will connect to unless appropriate network security measures are used to separate the traffic.

Fidelio driver

We have a customer interested in the Fidelio Driver. Is it still available?

Oracle acquired Micros. Their product portfolio refers to **Oracle Hospitality** rather than **Fidelio**. The Niagara Micros driver works with the legacy Micros Fidelio driver.

IFC Configuration

The OPERA IFC8 configuration requires a Product Code. What is this?

You may be required to either configure or assist in the configuration of the PMS via an IFC (Interface) Web Configuration utility. One of the entries in the IFC (Interface) window requires a **Product Code**. The **Product Code** of the Niagara Micros driver interface is **IO-5007–065**.

Large Installations

What do I need to consider when using the driver in a large installation?

For sites with 1000's of rooms, a Supervisor will be required to handle the volume of messages sent by the PMS and these points would typically be exported to room controllers via the Niagara Network using BACnet or another protocol. If you are planning to use a JACE with a Micros Driver per floor, please note that the FIAS protocol does not allow subscription to specific rooms. This will mean that each connected device will receive unsolicited traffic for room events across the entire site.

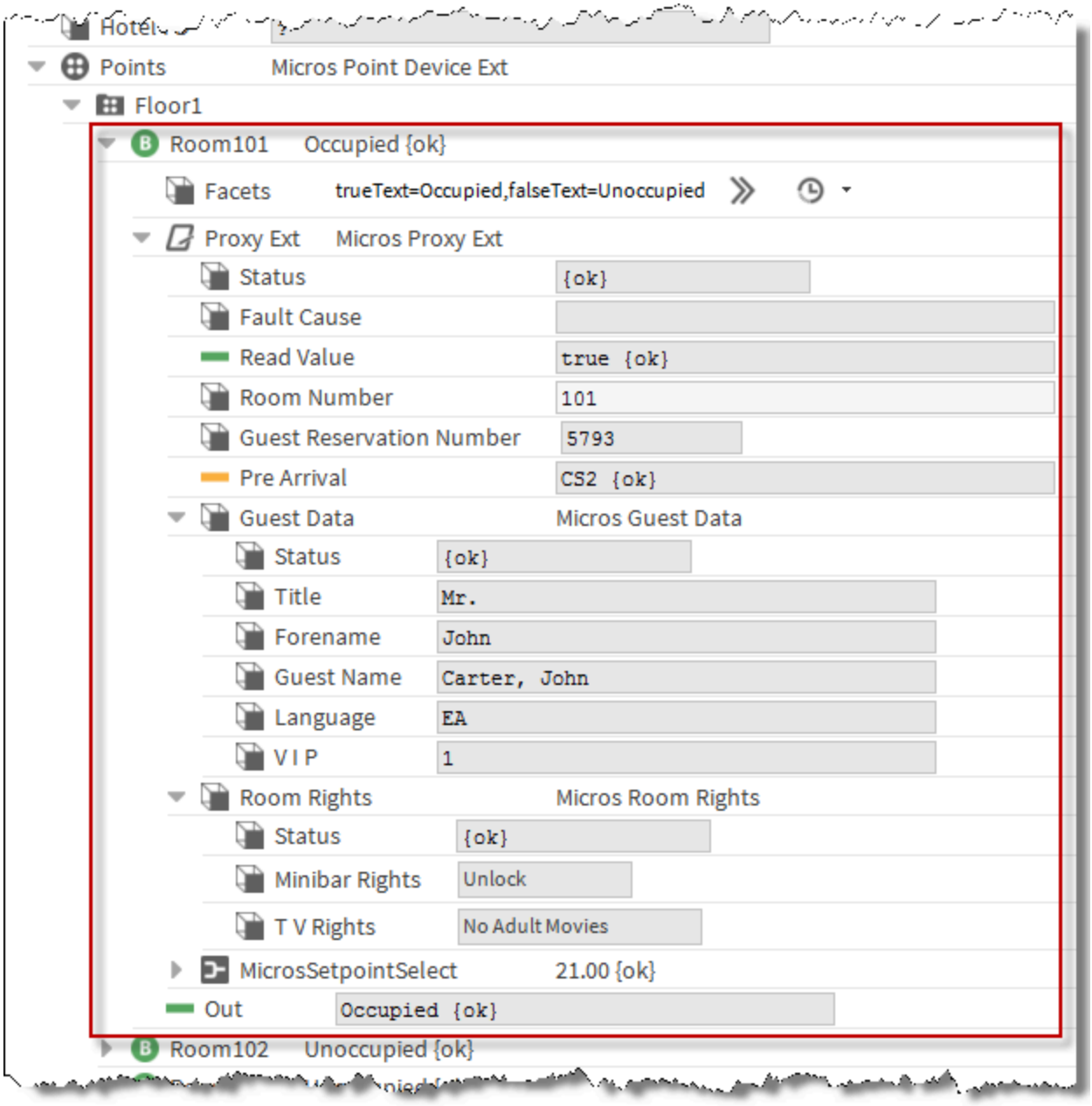
Personal Guest Data

I am concerned about the security of Personal Guest data?

The first and most important point to recognise is that the Niagara Micros driver is just a **listener** of Guest Room data traffic and it does not have any control over the amount nor the type of data transmitted by the PMS.

Configuration options in the Niagara Micros driver give the installation engineer the flexibility to choose what Guest Room data is made available in the station database.

The fundamental Guest Room data collected by the Niagara Micros driver from the PMS is **Room Number**, **Guest Reservation Number** and most importantly, whether the room is **Occupied** or not. These data **only** contain PMS Room Occupancy details and do not contain any Personal Guest data. These data are always automatically collected and are illustrated in the image below:



The collection of Personal Guest data including **Title** and **Name**, as well as **Pre-Arrival**, **TV** and **Minibar** Room Rights data are enabled or disabled by configuration settings in the Niagara Micros driver and are setup by the installation engineer. Personal guest data and Room Rights data are also illustrated in the image above.

Tiger TMS iCharge

Do you know if the driver is compatible with Tiger TMS iCharge?

Tiger TMS iCharge is 'middleware' software that has a PBX functionality interface to a Micros Opera PMS. Tiger TMS iCharge operates between a Micros Opera PMS and telephone systems, voice mail, high speed internet, iConnect (POS), iPGS (Guest Services), BMS systems and Niagara. To all these services and systems, Tiger TMS iCharge presents an FIAS emulation interface. We are aware that such an interface has been successfully tested with the Micros driver. However, because the Tiger TMS iCharge emulation presents PBX functionality it only supports Guest Check-in and Guest Check-out.

CAUTION: You should nevertheless, exercise caution and it is strongly recommended that before making any commitment to operate with a third party PMS system you review the Vendor Record Specification Form, compare this with the message types supported by the third party system and test compatibility with an example.

Assumed Checkout

A Micros driver term used to describe an unoccupied room

BMS

Building Management System

Database Sync

Database Sync is the process of synchronizing the data in the driver with what is currently on site in the PMS system

FIAS

Fidelio Interface Application Specification [the protocol]

Guest Check-in

An FIAS record used to transmit the check-in data of guests

Guest Check-out

An FIAS record used to transmit the check-out data of guests

Guest data change

An FIAS record used to transmit changed data (for example room change) of guests

IFC

Interface Server (Micros Software)

Oracle Micros

The vendor of FIAS

PMS

Property Management System [for example, Suite8, Opera]

Pre-Arrival

A FIAS Room Equipment (RE) record sent prior to guest arrival

VRSF

Vendor Record Specification Form (Protocol Conformance Statement)