iControl

Signal and facility monitoring

iControl Services Gateway Reference Guide

M226-0404-101

6 November 2014

grass valley
A BELDEN BRAND

Copyright and Trademark Notice

Warranty Policies

Warranty information is available in the Support section of the Miranda Web site (www.miranda.com).

Title	iControl Services Gateway Reference Guide
Part Number	M226-0404-101
Revision Date	6 November 2014 2:03 pm

toc

Table of Contents

iControl Services Gateway Reference

This manual is designed as a reference for programmers of third-party network management systems (NMS). The workflow and individual commands documented in this manual are meant to be integrated into driver software to enable direct communication—through the iControl Services Gateway—between the NMS and a Densité card or ImageStore device.

Notes

- The following workflow assumes that the controller is an NMS. However, for purposes of display, screen captures of human-operated *Telnet* sessions will be included to show command-by-command responses from the iControl Services Gateway as well as from an individual service node (i.e. a Densité card or an ImageStore card).
- The workflow steps involving XML commands—as well as those steps that require setting up Telnet sessions—would be coded directly into the driver if the controller is an NMS.

REQUIREMENTS

Make sure you meet the following conditions before beginning this workflow:

- You know the IP address of the Application Server whose iControl Services Gateway application you wish to connect to.
- **[IF APPLICABLE]** You are able to edit the driver software that will allow your network management system software to interact with the iControl Services Gateway.
- **[IF APPLICABLE]** You are able to launch Telnet sessions to the Application Server.

Sample workflow:

1.	Enable the iControl Services Gateway service on the appropriate Application Server (see page 2).
2.	Retrieve and open the XML file associated to the service node type whose parameters you would like to edit or retrieve (see page 2).
3.	Set up a Telnet session to the *iControl Gateway Directory Service* application of the Application Server (see "Operations performed on the gateway Directory Service", on page 4).
4.	Request a list of registered service nodes from the iControl Directory Services (see "Operations performed on the gateway Directory Service", on page 4).
5.	Close the session with the Directory Services.

Sample workflow: (*Continued*)

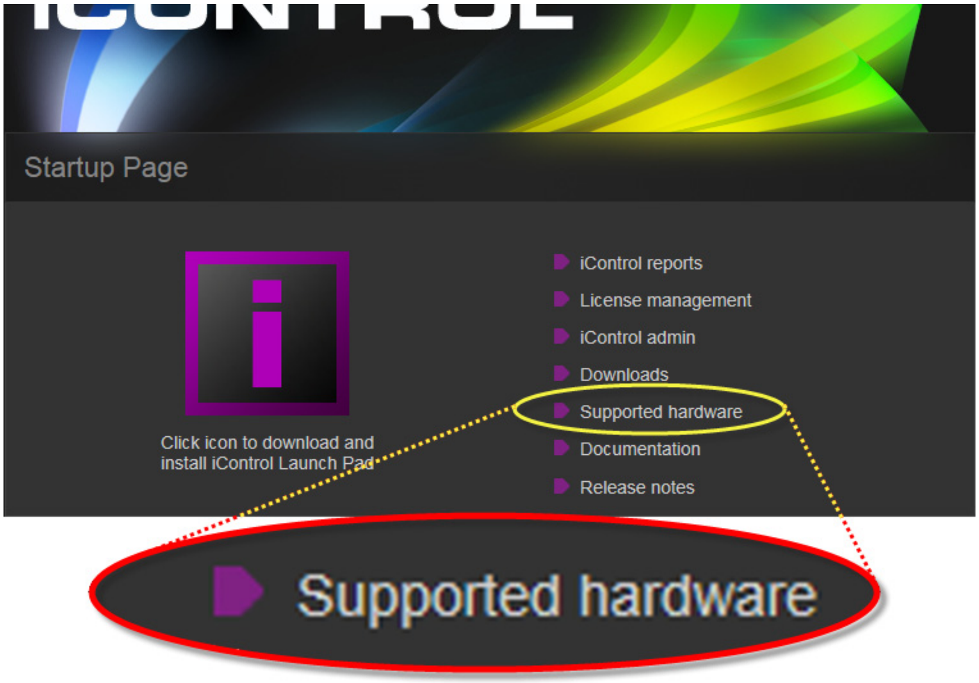
6.	Set up another Telnet session to the registered service node (e.g. a particular XVP-3901 Densité card on a particular frame) whose parameters you would like to edit.
7.	Edit or retrieve configuration data on the service node.
8.	Close the session with the service node.

Enabling the *iControl Services Gateway* service

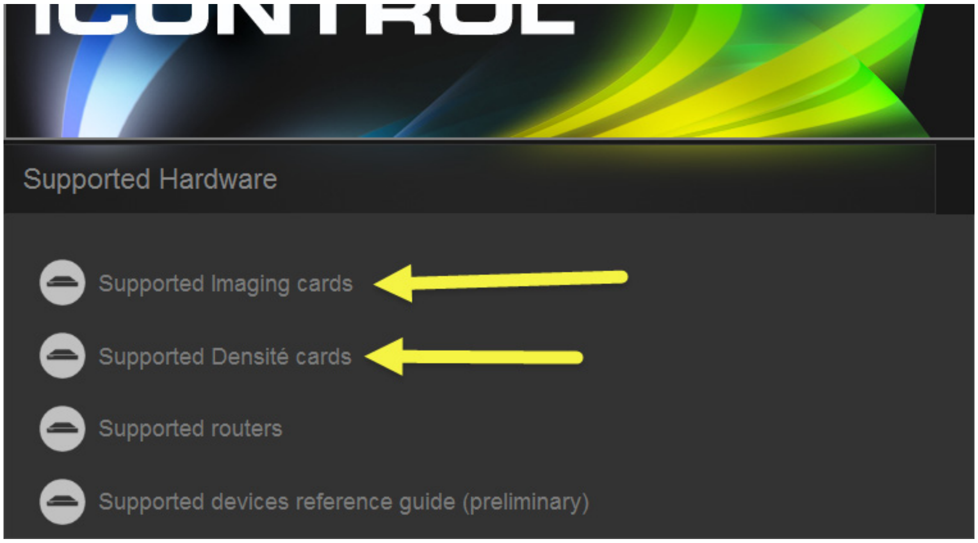
1. Open iControl in a browser.
2. On the *iControl—admin* page, click **Services management**.
3. Navigate to the bottom of the list of services.
4. In the **Autostart** column, click to put a check mark in the **Start** box corresponding to the **iControl Services Gateway** row.

Retrieving and opening a service node's iC Gateway Reference file

1. In iControl, on the *iControl—Startup* page, click **Supported hardware**.



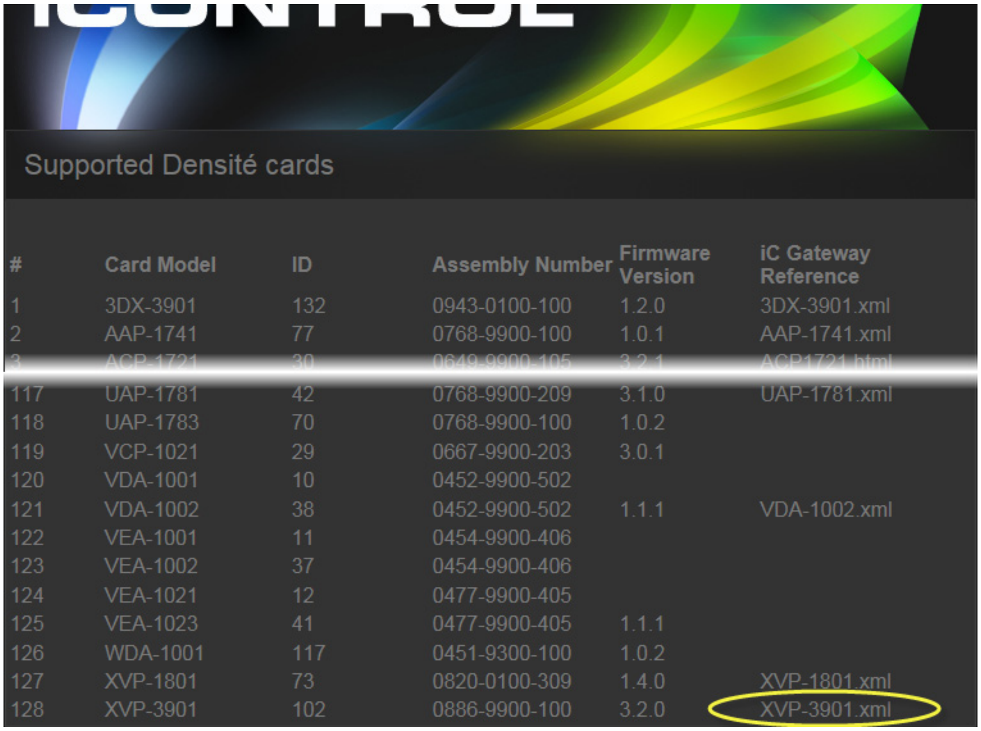
2. On the *iControl—Supported Hardware* page, click the link that best describes the service node type.



Supported Hardware

- Supported Imaging cards
- Supported Densité cards
- Supported routers
- Supported devices reference guide (preliminary)

3. Locate the service node type in the list, and then click the link for the associated XML file.

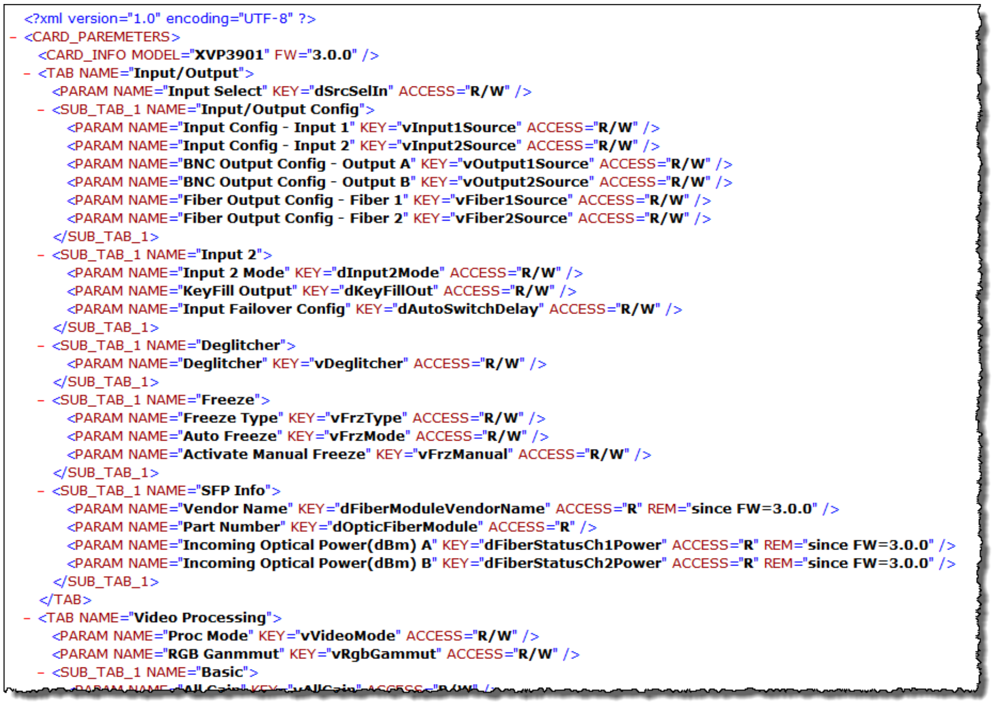


Supported Densité cards

#	Card Model	ID	Assembly Number	Firmware Version	iC Gateway Reference
1	3DX-3901	132	0943-0100-100	1.2.0	3DX-3901.xml
2	AAP-1741	77	0768-9900-100	1.0.1	AAP-1741.xml
3	ACP-1721	30	0649-9900-105	3.2.1	ACP1721.html
117	UAP-1781	42	0768-9900-209	3.1.0	UAP-1781.xml
118	UAP-1783	70	0768-9900-100	1.0.2	
119	VCP-1021	29	0667-9900-203	3.0.1	
120	VDA-1001	10	0452-9900-502		
121	VDA-1002	38	0452-9900-502	1.1.1	VDA-1002.xml
122	VEA-1001	11	0454-9900-406		
123	VEA-1002	37	0454-9900-406		
124	VEA-1021	12	0477-9900-405		
125	VEA-1023	41	0477-9900-405	1.1.1	
126	WDA-1001	117	0451-9300-100	1.0.2	
127	XVP-1801	73	0820-0100-309	1.4.0	XVP-1801.xml
128	XVP-3901	102	0886-9900-100	3.2.0	XVP-3901.xml

Link to the iC Gateway Reference (XML) file describing parameters for the XVP-3901 card

The XML file opens in your browser.



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <CARD_PAREMETERS>
    <CARD_INFO MODEL="XVP3901" FW="3.0.0" />
  - <TAB NAME="Input/Output">
      <PARAM NAME="Input Select" KEY="dSrcSelIn" ACCESS="R/W" />
    - <SUB_TAB_1 NAME="Input/Output Config">
        <PARAM NAME="Input Config - Input 1" KEY="vInput1Source" ACCESS="R/W" />
        <PARAM NAME="Input Config - Input 2" KEY="vInput2Source" ACCESS="R/W" />
        <PARAM NAME="BNC Output Config - Output A" KEY="vOutput1Source" ACCESS="R/W" />
        <PARAM NAME="BNC Output Config - Output B" KEY="vOutput2Source" ACCESS="R/W" />
        <PARAM NAME="Fiber Output Config - Fiber 1" KEY="vFiber1Source" ACCESS="R/W" />
        <PARAM NAME="Fiber Output Config - Fiber 2" KEY="vFiber2Source" ACCESS="R/W" />
      </SUB_TAB_1>
    - <SUB_TAB_1 NAME="Input 2">
        <PARAM NAME="Input 2 Mode" KEY="dInput2Mode" ACCESS="R/W" />
        <PARAM NAME="KeyFill Output" KEY="dKeyFillOut" ACCESS="R/W" />
        <PARAM NAME="Input Failover Config" KEY="dAutoSwitchDelay" ACCESS="R/W" />
      </SUB_TAB_1>
    - <SUB_TAB_1 NAME="Deglitcher">
        <PARAM NAME="Deglitcher" KEY="vDeglitcher" ACCESS="R/W" />
      </SUB_TAB_1>
    - <SUB_TAB_1 NAME="Freeze">
        <PARAM NAME="Freeze Type" KEY="vFrzType" ACCESS="R/W" />
        <PARAM NAME="Auto Freeze" KEY="vFrzMode" ACCESS="R/W" />
        <PARAM NAME="Activate Manual Freeze" KEY="vFrzManual" ACCESS="R/W" />
      </SUB_TAB_1>
    - <SUB_TAB_1 NAME="SFP Info">
        <PARAM NAME="Vendor Name" KEY="dFiberModuleVendorName" ACCESS="R" REM="since FW=3.0.0" />
        <PARAM NAME="Part Number" KEY="dOpticFiberModule" ACCESS="R" />
        <PARAM NAME="Incoming Optical Power(dBm) A" KEY="dFiberStatusCh1Power" ACCESS="R" REM="since FW=3.0.0" />
        <PARAM NAME="Incoming Optical Power(dBm) B" KEY="dFiberStatusCh2Power" ACCESS="R" REM="since FW=3.0.0" />
      </SUB_TAB_1>
    </TAB>
  - <TAB NAME="Video Processing">
      <PARAM NAME="Proc Mode" KEY="vVideoMode" ACCESS="R/W" />
      <PARAM NAME="RGB Ganmmut" KEY="vRgbGammut" ACCESS="R/W" />
    - <SUB_TAB_1 NAME="Basic">
        <PARAM NAME="All Gain" KEY="vAllGain" ACCESS="R/W" />
```

4. Keep this browser window open for reference. You will need the access keys in this `XML` file later.

Operations performed on the gateway Directory Service

REQUIREMENT

Before beginning this procedure, make sure you familiarize yourself with syntax conventions and the meanings of the variables in the examples (see "Syntax variables", on page 5).

1. Launch a Telnet session—using an application like *PuTTY*—to the *iControl Gateway Directory Service* by typing the following:

   `telnet%{appServerAddress}%10001`

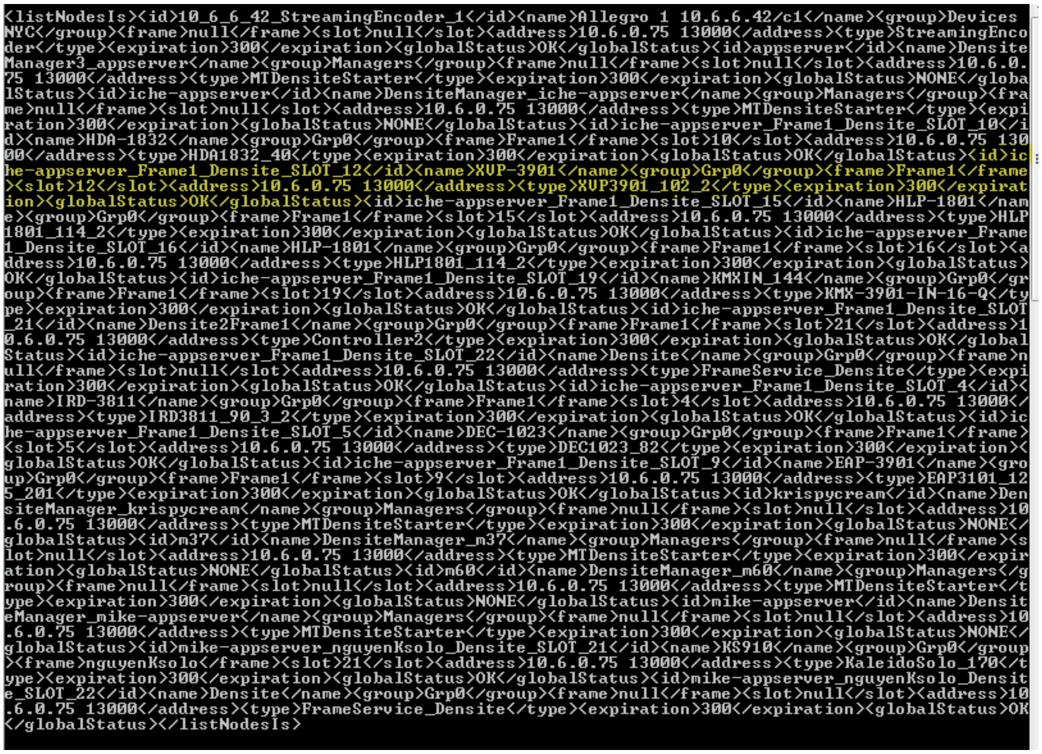
   (For example: `telnet 10.6.0.75 10001`)

2. Request a list of registered service nodes by typing the following:

   `<listNodes/>`

   The system returns a list of registered nodes (i.e. all Densité cards and ImageStore devices that are currently present in the iControl system and managed by this iControl server).

3. Locate the service node (e.g. a Densité card) in the list.



```
<listNodesIs><id>10_6_6_42_StreamingEncoder_1</id><name>Allegro 1 10.6.6.42/c1</name><group>Devices
NYC</group><frame>null</frame><slot>null</slot><address>10.6.0.75 13000</address><type>StreamingEnco
der</type><expiration>300</expiration><globalStatus>OK</globalStatus><id>appserver</id><name>Densite
Manager3_appserver</name><group>Managers</group><frame>null</frame><slot>null</slot><address>10.6.0.
75 13000</address><type>MTDensiteStarter</type><expiration>300</expiration><globalStatus>NONE</globa
lStatus><id>iche-appserver</id><name>DensiteManager_iche-appserver</name><group>Managers</group><fra
me>null</frame><slot>null</slot><address>10.6.0.75 13000</address><type>MTDensiteStarter</type><expi
ration>300</expiration><globalStatus>NONE</globalStatus><id>iche-appserver_Frame1_Densite_SLOT_10</i
d><name>HDA-1832</name><group>Grp0</group><frame>Frame1</frame><slot>10</slot><address>10.6.0.75 130
00</address><type>HDA1832_40</type><expiration>300</expiration><globalStatus>OK</globalStatus><id>ic
he-appserver_Frame1_Densite_SLOT_12</id><name>XVP-3901</name><group>Grp0</group><frame>Frame1</frame
><slot>12</slot><address>10.6.0.75 13000</address><type>XVP3901_102_2</type><expiration>300</expirat
ion><globalStatus>OK</globalStatus><id>iche-appserver_Frame1_Densite_SLOT_15</id><name>HLP-1801</nam
e><group>Grp0</group><frame>Frame1</frame><slot>15</slot><address>10.6.0.75 13000</address><type>HLP
1801_114_2</type><expiration>300</expiration><globalStatus>OK</globalStatus><id>iche-appserver_Frame
1_Densite_SLOT_16</id><name>HLP-1801</name><group>Grp0</group><frame>Frame1</frame><slot>16</slot><a
ddress>10.6.0.75 13000</address><type>HLP1801_114_2</type><expiration>300</expiration><globalStatus>
OK</globalStatus><id>iche-appserver_Frame1_Densite_SLOT_19</id><name>KMXIN_144</name><group>Grp0</gr
oup><frame>Frame1</frame><slot>19</slot><address>10.6.0.75 13000</address><type>KMX-3901-IN-16-Q</ty
pe><expiration>300</expiration><globalStatus>OK</globalStatus><id>iche-appserver_Frame1_Densite_SLOT
_21</id><name>Densite2Frame1</name><group>Grp0</group><frame>Frame1</frame><slot>21</slot><address>1
0.6.0.75 13000</address><type>Controller2</type><expiration>300</expiration><globalStatus>OK</global
Status><id>iche-appserver_Frame1_Densite_SLOT_22</id><name>Densite</name><group>Grp0</group><frame>n
ull</frame><slot>null</slot><address>10.6.0.75 13000</address><type>FrameService_Densite</type><expi
ration>300</expiration><globalStatus>OK</globalStatus><id>iche-appserver_Frame1_Densite_SLOT_4</id><
name>IRD-3811</name><group>Grp0</group><frame>Frame1</frame><slot>4</slot><address>10.6.0.75 13000</
address><type>IRD3811_90_3_2</type><expiration>300</expiration><globalStatus>OK</globalStatus><id>ic
he-appserver_Frame1_Densite_SLOT_5</id><name>DEC-1023</name><group>Grp0</group><frame>Frame1</frame>
<slot>5</slot><address>10.6.0.75 13000</address><type>DEC1023_82</type><expiration>300</expiration><
globalStatus>OK</globalStatus><id>iche-appserver_Frame1_Densite_SLOT_9</id><name>EAP-3901</name><gro
up>Grp0</group><frame>Frame1</frame><slot>9</slot><address>10.6.0.75 13000</address><type>EAP3101_12
5_201</type><expiration>300</expiration><globalStatus>OK</globalStatus><id>krispycream</id><name>Den
siteManager_krispycream</name><group>Managers</group><frame>null</frame><slot>null</slot><address>10
.6.0.75 13000</address><type>MTDensiteStarter</type><expiration>300</expiration><globalStatus>NONE</
globalStatus><id>m37</id><name>DensiteManager_m37</name><group>Managers</group><frame>null</frame><s
lot>null</slot><address>10.6.0.75 13000</address><type>MTDensiteStarter</type><expiration>300</expir
ation><globalStatus>NONE</globalStatus><id>m60</id><name>DensiteManager_m60</name><group>Managers</g
roup><frame>null</frame><slot>null</slot><address>10.6.0.75 13000</address><type>MTDensiteStarter</t
ype><expiration>300</expiration><globalStatus>NONE</globalStatus><id>mike-appserver</id><name>Densit
eManager_mike-appserver</name><group>Managers</group><frame>null</frame><slot>null</slot><address>10
.6.0.75 13000</address><type>MTDensiteStarter</type><expiration>300</expiration><globalStatus>NONE</
globalStatus><id>mike-appserver_nguyenKsolo_Densite_SLOT_21</id><name>KS910</name><group>Grp0</group
><frame>nguyenKsolo</frame><slot>21</slot><address>10.6.0.75 13000</address><type>KaleidoSolo_170</t
ype><expiration>300</expiration><globalStatus>OK</globalStatus><id>mike-appserver_nguyenKsolo_Densit
e_SLOT_22</id><name>Densite</name><group>Grp0</group><frame>null</frame><slot>null</slot><address>10
.6.0.75 13000</address><type>FrameService_Densite</type><expiration>300</expiration><globalStatus>OK
</globalStatus></listNodesIs>
```

Information about single service node (an XVP-3901 card) highlighted

4. Make note of the service node's identifier (ID), found between the `<id>` and `</id>` tags.

   In this example, the service node ID is `iche-appserver_Frame1_Densite_SLOT_12`.

5. Launch a different Telnet session to the service node by typing the following:

   `telnet%{appServerAddress}%13000`

6. In the new Telnet session, open a session to the service node by typing the following:

   `<openID>{serviceNodeID}</openID>`

   The system returns `<ack/>` if the session launched successfully, or `<nack/>` if not.

7. Perform operations on the service node's configuration as desired (see "Operations performed on a service node", on page 6).

Syntax variables

Variable	Notes
`%`	Indicates a <**space**> character
{appServerAddress}[1]	IPv4 address of the Application Server
{serviceNodeID}	Identifier of the service node (e.g. a Densité card) whose parameters you would like to retrieve or configure. The value of this identifier is retrievable with the `<listNodes>` command (see "Operations performed on the gateway Directory Service", on page 4).

1. The notation in this manual of placing variable names within **curly** brackets (e.g. `{variable}`), is to provide more visual contrast from syntax that should be entered exactly as indicated. When typing commands, **DO NOT** include the curly brackets.

Operations performed on a service node

Command: <**getParameterInfo**>.

Purpose

To retrieve all information (available to an application) about the parameters of an access key.

Command syntax

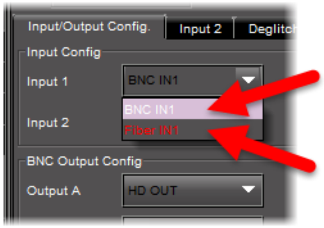
```
<getParameterInfo%key=”{key}”/>
```

Response syntax—Parameters with discrete ‘choices’

[Success scenario—all values accepted]

```
<{key}>{paramValue}</{key}>
```

```
<parameterInfo%name=”{paramName}”%type=”{paramType}”%isActive=”{paramActive}><
choice%label=”{choiceLabel}”%rcpval=”{choiceRcpVal}”%active=”{choiceActive}/>
```



GUI control panel parameter showing discrete choices in a combo box

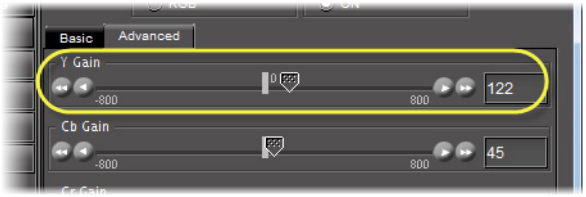
[Fail scenario—NOT all values accepted]

```
<nack/>
```

Response syntax—Parameters with continuous range of values

```
<{key}>{paramValue}</{key}>
```

```
<parameterInfo%name=”{paramName}”%isActive=”{paramActive}”%min=”{paramRangeMin
}”%max=”{paramRangeMax}”%step=”{paramRangeStep}”%fstep=”{paramRangeFstep}”%nom
inal=”{paramRangeNominal}”%unit=”{paramUnit}”/>
```



GUI control panel parameter showing continuous range (i.e. slider)

Syntax variables

Variable	Notes
`%`	Indicates a <**space**> character
{key}[1]	Access key Available from the *iC Gateway Reference* XML file you opened in your browser (see step 4 of "Retrieving and opening a service node's iC Gateway Reference file" on page 2).
{paramValue}	Current value of the parameter
{paramName}	Parameter name
{paramType}	Parameter type
{paramActive}	
{choiceLabel}	Label (**not** the value) identifying a choice
{choiceRcpVal}	Value that this choice would give to the parameter if chosen.
{choiceActive}	
{paramRangeMin}	
{paramRangeMax}	
{paramRangeStep}	
{paramRangeFstep}	
{paramRangeNominal}	
{paramUnit}	

[1] The notation in this manual of placing variable names within **curly** brackets (e.g. *{variable}*), is to provide more visual contrast from syntax that should be entered exactly as indicated. When typing commands, **DO NOT** include the curly brackets.

Example—Parameter with 'choice' values

User input

```
<getParameterInfo key=”vThumb_Q”/>
```

System response

```
<vThumb_Q>Normal</vThumb_Q>
```

```
<parameterInfo name=”Quality” type=”choice” isActive=”true”><choice
label=”Poor” rcpval=”Poor” active=”true”/><choice label =”Normal”
rcpval=”Normal” active=”true”/><choice label =”HiQ” rcpval=”HiQ”
active=”true”/></parameterInfo>
```

Note: The “`choice`” value returned by the system for `type`, as seen above, indicates that the data type is *enum* (*enumerated*), meaning there are preset values in a discrete set. In the above example, there are three possible values: **`Poor`**, **`Normal`**, and **`HiQ`**.

Example—Parameter with continuous range of possible values

User input

```
<getParameterInfo key=”vLuma”/>
```

System response

```
<vLuma>122 </vLuma>
```

```
<parameterInfo name=”Y Gain” isActive=”true” min=”-800.0” max=”800.0”
step=”1.0” fstep=”25” nominal=”0.0” unit=””/>
```

Command: **<get[Parameter]>**

Purpose

To retrieve the value associated with a parameter of an access key.

Command syntax

`<get{key1stLetterUpperCase}/>`

Response syntax

`<{key}>{paramValue}</{key}>`

Syntax variables

variable	Notes
{key}[1]	Access key Available from the *iC Gateway Reference* `XML` file you opened in your browser (see step 4 of "Retrieving and opening a service node’s iC Gateway Reference file" on page 2).
{key1stLetterUpperCase}	Access key with its first character in upper case For example, if an access key is **dAFDMode2**, the *{key1stLetterUpperCase}* string would be **DAFDMode2**.
{paramValue}	Current value of the parameter

[1] The notation in this manual of placing variable names within **curly** brackets (e.g. *{variable}*), is to provide more visual contrast from syntax that should be entered exactly as indicated. When typing commands, **DO NOT** include the curly brackets.

Example

User input

```
<getDAFDMode2/>
```

System response

```
<dAFDMode2>4:3 </dAFDMode2>
```

Command: **<set[Parameter]>**

Purpose

To configure the value of a service node parameter.

Command syntax

```
<set{key1stLetterUpperCase}/>{type}%{value}</set{key1stLetterUpperCase}>
```

Response syntax

[Success scenario—all values accepted]

```
<ack/>
```

```
<{key}>{paramValue}</{key}>
```

> **Note:** There may be several lines of response, giving several parameter values. The system returns those parameter values that are affected by this SET command. The example, below, is one in which several parameters were affected by setting one parameter.

[Fail scenario—NOT all values accepted]

```
<nack/>
```

Syntax variables

Variable	**Notes**
%	Indicates a <**space**> character
{key}[1]	Access key Available from the *iC Gateway Reference* XML file you opened in your browser (see step 4 of "Retrieving and opening a service node's iC Gateway Reference file" on page 2).
{key1stLetterUpperCase}	Access key with its first character in upper case For example, if an access key is **dAFDMode2**, the *{key1stLetterUpperCase}* string would be **DAFDMode2**.
{paramValue}	Actual value of the parameter

(*Continued*)

Variable	Notes
{type}	Indicates the action-type you would like to perform on the value of the parameter. • **inc**—Increments the current value by the indicated quantity • **dec**—Decrements the current value by the indicated quantity • **set**—Configures the value of the parameter precisely to the indicated quantity
{value}	Value to set a parameter to, or increment by, or decrement by

1. The notation in this manual of placing variable names within **curly** brackets (e.g. *{variable}*), is to provide more visual contrast from syntax that should be entered exactly as indicated. When typing commands, **DO NOT** include the curly brackets.

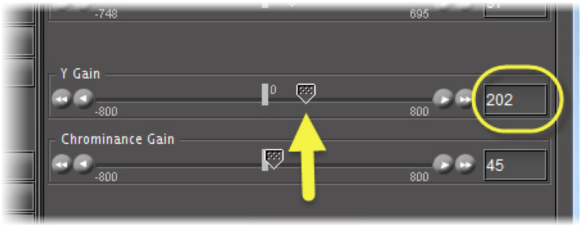
Example

User input

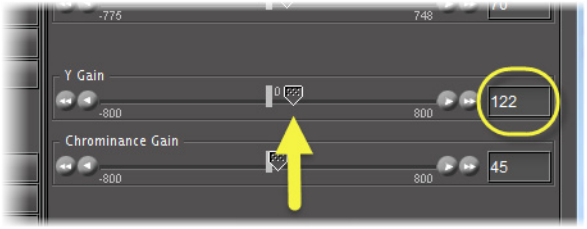
```
<setVLuma>dec 80</setVLuma>
```

System response

```
<ack/>

<vLuma>122 </vLuma>

<vAllGain>70 </vAllGain>

<vYGain>122 </vYGain>

<vAllGain>70 </vAllGain>
```



*GUI control panel **BEFORE** sending SET command*



*GUI control panel **AFTER** sending SET command*

Command: <closeID>

Purpose

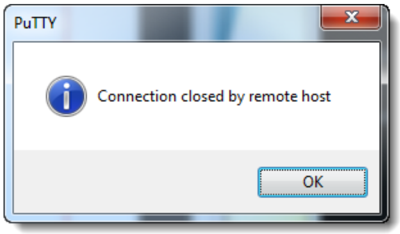
To close a Telnet session with a service node.

Command syntax

```
<closeID>{ID}</closeID>
```

Response syntax

If you are using *PuTTY* and your session is configured not to close upon exiting, the following window appears.



Connection closed by remote host

Syntax variables

Variable	Notes
`{ID}`[1]	Identifier of the service node (e.g. a Densité card) Retrieved through the `<listNodes>` command (see "Operations performed on the gateway Directory Service", on page 4).

1. The notation in this manual of placing variable names within **curly** brackets (e.g. `{variable}`), is to provide more visual contrast from syntax that should be entered exactly as indicated. When typing commands, **DO NOT** include the curly brackets.

Example

User input

```
<closeID>iche-appserver_Frame1_Densite_SLOT_12</closeID>
```

Contact Us

Grass Valley Technical Support

For technical assistance, please contact the *Grass Valley Technical Support* center nearest you:

Americas

Office hours:	9:00 a.m. - 9:00 p.m. (EST)
Telephone:	1-800-224-7882
Fax:	+1 514 335 1614
E-mail:	`support@grassvalley.com`

Europe, Middle East, Africa, UK

Office hours:	9:00 a.m. - 6:00 p.m. (GMT)
Telephone:	+ 44 118 952 3444
Fax:	+ 44 118 952 3401
E-mail:	`eurotech@grassvalley.com`

France

Office hours:	9:00 a.m. - 5:00 p.m. (GMT + 1)
Telephone:	+ 33 1 55 86 87 88
Fax:	+ 33 1 55 86 00 29
E-mail:	`eurotech@grassvalley.com`

Asia

Office hours:	9:00 a.m. - 5:00 p.m. (GMT+8)
Telephone:	+ 852 2539 6987
Fax:	+ 852 2539 0804
E-mail:	`asiatech@grassvalley.com`

China

Telephone:	+ 86 10 5873 1814
E-mail:	`asiatech@grassvalley.com`

EMERGENCY After Hours (Global)

Toll Free:	1-800-224-7882 (US and Canada)
Telephone:	+1 514 333 1772

Corporate Head Office

Grass Valley, a Belden Brand
3499 Douglas-B.-Floreani, St-Laurent, Québec, Canada H4S 2C6

Telephone:	+514 333 1772
Fax:	+514 333 9828
Web:	`www.grassvalley.com`