

# **ICONTROL GSM REST API**

## **Reference Guide**

M3059-0104-100

2018-02-16

## Copyright and Trademark Notice

Copyright © 2017-2018, Grass Valley Canada. All rights reserved.

Belden, Belden Sending All The Right Signals, and the Belden logo are trademarks or registered trademarks of Belden Inc. or its affiliated companies in the United States and other jurisdictions. Grass Valley, Densité, GV Node, GeckoFlex, and iControl are trademarks or registered trademarks of Grass Valley Canada. Belden Inc., Grass Valley Canada, and other parties may also have trademark rights in other terms used herein.

## Terms and Conditions

Please read the following terms and conditions carefully. By using iControl Application Server documentation, you agree to the following terms and conditions.

Grass Valley hereby grants permission and license to owners of iControl Application Servers to use their product manuals for their own internal business use. Manuals for Grass Valley products may not be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, for any purpose unless specifically authorized in writing by Grass Valley.

A Grass Valley manual may have been revised to reflect changes made to the product during its manufacturing life. Thus, different versions of a manual may exist for any given product. Care should be taken to ensure that one obtains the proper manual version for a specific product serial number.

Information in this document is subject to change without notice and does not represent a commitment on the part of Grass Valley.

Warranty information is available from the Legal Terms and Conditions section of Grass Valley's website ([www.grassvalley.com](http://www.grassvalley.com)).

Title	iControl GSM REST API Reference Guide
Part Number	M3059-0104-100
Revision	2018-02-16, 15:54

# toc

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>1</b>
<b>2</b>	<b>Service description .....</b>	<b>3</b>
	Individual requests .....	4
	Get the WADL file .....	5
	Get the REST API version .....	6
	Get a specific alarm using an alarm URI .....	7
	Get specific alarm folders using alarmFolderDto .....	8
	Get a specific alarm folder using alarm folder URI .....	9
	Set the status of a specific alarm .....	10
	Get the root alarm folder .....	11
	Session based requests .....	12
	Register a session for the GSM Rest API .....	13
	Register a session .....	14
	Disconnect a session .....	15
	Register a session for listening on a specific folder .....	16
	Add an alarm .....	17
	Remove alarms .....	18
	Unregister listening on a specific alarm .....	19
	Unregister listening on a specific folder .....	20
	Register listening on a specific alarm .....	21
	Dispatch alarms .....	22
<b>3</b>	<b>JSON objects .....</b>	<b>23</b>
	AlarmDescriptorDto .....	24
	AlarmDescriptorListDto .....	25
	AlarmStateDto .....	26
	AlarmStatesSnapshotDto .....	27
	AlarmDto .....	28
	AlarmFolderDto .....	29
	AlarmFolderEntryListDto .....	30
	DispatchDto .....	31



# 1 Introduction

This document describes iControl's GSM REST service. This service requires the *iControl Services Gateway* license (order code: IC-GATEWAY).

As of iControl version 7.20, the GSM provides a REST service to access alarm status.

The REST service is activated with the following configuration:

- Available via HTTP on the iControl Application Server
- Registered on port:
  - 5957
- With base path:
  - `http://10.0.3.6:5957/gsm`

A WADL file describing the service is available in the base path:

- `http://10.0.3.6:5957/gsm/application.wadl`

---

**Note:** In this document, the IP address `10.0.3.6` represents the iControl Application Server.

---



# 2

## Service description

This chapter provides the service description for the GSM REST API.

### Summary

[Individual requests](#), on page 4

[Session based requests](#), on page 12

---

## Individual requests

### Summary

[Get the WADL file, on page 5](#)

[Get the REST API version, on page 6](#)

[Get a specific alarm using an alarm URI, on page 7](#)

[Get specific alarm folders using alarmFolderDto, on page 8](#)

[Set the status of a specific alarm, on page 10](#)

[Get the root alarm folder, on page 11](#)

[Session based requests, on page 12](#)



## Get the WADL file

Path	http://10.0.3.6:5957/gsm/application.wadl
Method	GET
Request format	N/A
Response	application.wadl
Response media type	text/plain

### HTTP request example

```
GET/gsm/application.wadl HTTP/1.1
```

```
Host: 10.0.3.6:5957
```

```
Cache-Control: no-cache
```

### HTTP response example

A list of all the endpoints and their required parameters.

## Get the REST API version

<b>Path</b>	<code>&lt;resources path="/version"&gt;</code>
<b>Method</b>	GET
<b>Request</b>	None
<b>Response</b>	<code>&lt;response&gt;</code> <code>&lt;representation mediaType="application/json"/&gt;</code> <code>&lt;/response&gt;</code>

## Get a specific alarm using an alarm URI

Path	<resource path="/alarms/alarm/">
Method	GET
Request	<request> <param name="Authorization" style="header" type="xs:string"/> </request>
Path parameters	<param name="id" style="query" type="xs:string"/>
Response	<response> <ns2:representation element="alarmDto" mediaType="application/json"/> </response>

### HTTP request example

```
GET
/gsm/alarms/alarm?id=snmp://appserver_VM_2:D9032:10.37.49.103/veVideoInputType
```

### HTTP response example

```
{
  "descriptor": {
    "uri": "snmp://appserver_VM_2:D9032:10.37.49.103/veVideoInputType",
    "type": 0
  },
  "timestamp": 0,
  "timecode": -1,
  "gsmTimestamp": 1470741930662,
  "flags": 0,
  "state": {
    "current": {
      "state": -3,
      "latched": -3,
      "acknowledged": -3,
      "opMode": 0
    },
    "previous": {
      "state": -3,
      "latched": -3,
      "acknowledged": -3,
      "opMode": 0
    }
  }
}
```

## Get specific alarm folders using alarmFolderDto

<b>Path</b>	<resource path="/alarms/folder">
<b>Method</b>	GET
<b>Request</b>	<pre>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmFolderDto" mediaType="application/json"/&gt; &lt;/request&gt;</pre>
<b>Response</b>	<pre>&lt;response&gt; &lt;ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/&gt; &lt;/response&gt;</pre>

### HTTP request example

```
{
  "path": [
    "iControl alarms"
  ]
}
```

### HTTP response example

```
{
  "folder": {
    "path": [
      "iControl alarms"
    ]
  },
  "subFolders": [
    {
      "path": [
        "iControl alarms",
        "Health monitoring"
      ],
      "descriptor": {
        "deviceUri": "folder",
        "deviceClass": "Folder",
        "name": "Health monitoring",
        "uri": "folderStatus://iControl+alarms/Health+monitoring",
        "type": 0
      }
    }
  ]
}
```

## Get a specific alarm folder using alarm folder URI

Path	<resource path="/alarms/folder">
Method	GET
Request	<request> <param name="Authorization" style="header" type="xs:string"/> </request>
Path parameters	<param name="id" style="query" type="xs:string"/> <b>Note: If an ID is not provided, the command returns the alarm root folder.</b>
Response	<response> <ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/> </response>

### HTTP request example

GET /alarms/folder?id=folderStatus://iControl+alarms/Health+monitoring

### HTTP response example

```
{
  "folder": {
    "path": [
      "iControl alarms"
    ]
  },
  "subFolders": [
    {
      "path": [
        "iControl alarms",
        "Health monitoring"
      ],
      "descriptor": {
        "deviceUri": "folder",
        "deviceClass": "Folder",
        "name": "Health monitoring",
        "uri": "folderStatus://iControl+alarms/Health+monitoring",
        "type": 0
      }
    }
  ]
}
```

## Set the status of a specific alarm

<b>Path</b>	<code>&lt;resource path="/alarm/mode/set"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmScheduleRequestDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Response</b>	<code>No response</code>

## Get the root alarm folder

Path	<resource path="/alarms/root">
Method	GET
Request	<request> <param name="Authorization" style="header" type="xs:string"/> </request>
Response	<response> <ns2:representation element="alarmFolderDto" mediaType="application/json"/> </response>

### HTTP response example

```
{
  "subFolders": [
    {
      "path": [
        "iControl alarms"
      ],
      "descriptor": {
        "deviceUri": "folder",
        "deviceClass": "Folder",
        "name": "iControl alarms",
        "uri": "folderStatus://iControl+alarms",
        "type": 0
      }
    }
  ]
}
```

## Session based requests

Sessions allow receiving notifications of changed parameters on specific cards. The content of the session output buffer is retrieved by a long-polling method. If the buffer contains data, this method returns immediately with the data. If not, then the call is held until data is acquired or a delay of 30 seconds is elapsed.

### Summary

[Register a session for the GSM Rest API, on page 13](#)

[Register a session, on page 14](#)

[Disconnect a session, on page 15](#)

[Register a session for listening on a specific folder, on page 16](#)

[Add an alarm, on page 17](#)

[Remove alarms, on page 18](#)

[Unregister listening on a specific alarm, on page 19](#)

[Unregister listening on a specific folder, on page 20](#)

[Register listening on a specific alarm, on page 21](#)

[Dispatch alarms, on page 22](#)



## Register a session for the GSM Rest API

<b>Path</b>	<code>&lt;resource path="session/connect"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="sessionConfiguration" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="gsmSessionDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>

## Register a session

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/folder/register"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmFolderDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>

## Disconnect a session

Path	<code>&lt;resource path="session/{sessionId}/disconnect"&gt;</code>
Method	<b>POST</b>
Request	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;/request&gt;</code>
Path parameters	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
Response	<b>No response</b>

## Register a session for listening on a specific folder

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/folder/register"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmFolderDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>

## Add an alarm

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/alarms/add"&gt;</code>
<b>Method</b>	<b>POST</b>
<b>Request</b>	<pre>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmDescriptorListDto" mediaType="application/json"/&gt; &lt;/request&gt;</pre>
<b>Path parameters</b>	<pre>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</pre>
<b>Response</b>	<b>No response</b>

---

## Remove alarms

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/alarms/remove"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmDescriptorListDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>

## Unregister listening on a specific alarm

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/alarms/unregister"&gt;</code>
<b>Method</b>	<b>POST</b>
<b>Request</b>	<pre>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmDescriptorListDto" mediaType="application/json"/&gt; &lt;/request&gt;</pre>
<b>Path parameters</b>	<pre>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</pre>
<b>Response</b>	<b>No response</b>

## Unregister listening on a specific folder

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/folder/unregister"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmFolderDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="alarmFolderEntryListDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>



## Register listening on a specific alarm

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/alarms/register"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmDescriptorListDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>&lt;response&gt; &lt;ns2:representation element="alarmListDto" mediaType="application/json"/&gt; &lt;/response&gt;</code>

---

## Dispatch alarms

<b>Path</b>	<code>&lt;resource path="session/{sessionId}/alarms/dispatch"&gt;</code>
<b>Method</b>	<code>POST</code>
<b>Request</b>	<code>&lt;request&gt; &lt;param name="Authorization" style="header" type="xs:string"/&gt; &lt;ns2:representation element="alarmListDto" mediaType="application/json"/&gt; &lt;/request&gt;</code>
<b>Path parameters</b>	<code>&lt;param name="sessionId" style="template" type="xs:string"/&gt;</code>
<b>Response</b>	<code>No response</code>

# 3

## JSON objects

### Summary

[AlarmDescriptorDto](#), on page 24

[AlarmStateDto](#), on page 26

[AlarmStatesSnapshotDto](#), on page 27

[AlarmDto](#), on page 28

[AlarmFolderDto](#), on page 29

[AlarmFolderEntryListDto](#), on page 30

[DispatchDto](#), on page 31

## AlarmDescriptorDto

```
{
  "alarmDescriptorDto" : {
    "deviceClass" : "device class",
    "deviceUri" : "device uri",
    "name" : "device name",
    "path" : [ "icontrol", "folder 1", "folder 2" ],
    "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
    "type" : 1
  }
}
```

## AlarmDescriptorListDto

```
{
  "alarmDescriptorListDto" : {
    "descriptors" : [ {
      "deviceClass" : "device class",
      "deviceUri" : "device uri",
      "name" : "device name",
      "path" : [ "icontrol", "folder 1", "folder 2" ],
      "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
      "type" : 1
    } ]
  }
}
```

## AlarmStateDto

```
{
  "alarmStateDto" : {
    "acknowledged" : 1,
    "latched" : 1,
    "opMode" : 1,
    "state" : 0
  }
}
```

## AlarmStatesSnapshotDto

```
{
  "alarmStatesSnapshotDto" : {
    "current" : {
      "acknowledged" : 1,
      "latched" : 1,
      "opMode" : 1,
      "state" : 0
    }
  }
}
```

---

## AlarmDto

```
{
  "alarmDto" : {
    "descriptor" : {
      "deviceClass" : "device class",
      "deviceUri" : "device uri",
      "name" : "device name",
      "path" : [ "icontrol", "folder 1", "folder 2" ],
      "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
      "type" : 1
    },
    "extraInfo" : "Extra info",
    "flags" : 1,
    "gsmTimestamp" : 1470930730364,
    "rootCauses" : [ {
      "descriptor" : {
        "deviceClass" : "device class",
        "deviceUri" : "device uri",
        "name" : "device name",
        "path" : [ "icontrol", "folder 1", "folder 2" ],
        "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
        "type" : 1
      },
      "extraInfo" : "Extra info",
      "flags" : 1,
      "gsmTimestamp" : 1470930730364,
      "timecode" : -1,
      "timestamp" : 0
    } ],
    "state" : {
      "current" : {
        "acknowledged" : 1,
        "latched" : 1,
        "opMode" : 1,
        "state" : 0
      }
    },
    "timecode" : -1,
    "timestamp" : 0
  }
}
```



## AlarmFolderDto

```
{
  "alarmFolderDto" : {
    "descriptor" : {
      "deviceClass" : "Folder",
      "deviceUri" : "folder",
      "name" : "folder 2",
      "type" : 0,
      "uri" : "folderStatus://iControl+alarms/icontrol/folder+1/folder+2"
    },
    "path" : [ "icontrol", "folder 1", "folder 2" ]
  }
}
```

---

## AlarmFolderEntryListDto

```
{
  "alarmFolderEntryListDto" : {
    "descriptors" : [ {
      "deviceClass" : "device class",
      "deviceUri" : "device uri",
      "name" : "device name",
      "path" : [ "icontrol", "folder 1", "folder 2" ],
      "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
      "type" : 1
    } ],
    "folder" : {
      "descriptor" : {
        "deviceClass" : "Folder",
        "deviceUri" : "folder",
        "name" : "folder 2",
        "type" : 0,
        "uri" :
"folderStatus://iControl+alarms/icontrol/folder+1/folder+2"
      },
      "path" : [ "icontrol", "folder 1", "folder 2" ]
    },
    "subFolders" : [ {
      "descriptor" : {
        "deviceClass" : "Folder",
        "deviceUri" : "folder",
        "name" : "folder 2",
        "type" : 0,
        "uri" :
"folderStatus://iControl+alarms/icontrol/folder+1/folder+2"
      },
      "path" : [ "icontrol", "folder 1", "folder 2" ]
    } ]
  }
}
```

## DispatchDto

```

{
  "dispatchDto" : {
    "alarms" : [ {
      "descriptor" : {
        "deviceClass" : "device class",
        "deviceUri" : "device uri",
        "name" : "device name",
        "path" : [ "icontrol", "folder 1", "folder 2" ],
        "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
        "type" : 1
      },
      "extraInfo" : "Extra info",
      "flags" : 1,
      "gsmTimestamp" : 1470930730364,
      "rootCauses" : [ {
        "descriptor" : {
          "deviceClass" : "device class",
          "deviceUri" : "device uri",
          "name" : "device name",
          "path" : [ "icontrol", "folder 1", "folder 2" ],
          "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
          "type" : 1
        },
        "extraInfo" : "Extra info",
        "flags" : 1,
        "gsmTimestamp" : 1470930730364,
        "timecode" : -1,
        "timestamp" : 0
      } ],
      "state" : {
        "current" : {
          "acknowledged" : 1,
          "latched" : 1,
          "opMode" : 1,
          "state" : 0
        }
      },
      "timecode" : -1,
      "timestamp" : 0
    } ],
    "folders" : [ {
      "descriptors" : [ {
        "deviceClass" : "device class",
        "deviceUri" : "device uri",
        "name" : "device name",
        "path" : [ "icontrol", "folder 1", "folder 2" ],
        "sortPath" : [ "icontrol", "folder 1", "folder 2" ],
        "type" : 1
      } ],
      "folder" : {
        "descriptor" : {
          "deviceClass" : "Folder",
          "deviceUri" : "folder",
          "name" : "folder 2",
          "type" : 0,

```

```
        "uri" :
"folderStatus://iControl+alarms/iconcontrol/folder+1/folder+2"
    },
    "path" : [ "iconcontrol", "folder 1", "folder 2" ]
},
"subFolders" : [ {
    "descriptor" : {
        "deviceClass" : "Folder",
        "deviceUri" : "folder",
        "name" : "folder 2",
        "type" : 0,
        "uri" :
"folderStatus://iControl+alarms/iconcontrol/folder+1/folder+2"
    },
    "path" : [ "iconcontrol", "folder 1", "folder 2" ]
} ]
} ]
}
}
```





## **Grass Valley Technical Support**

For technical assistance, contact our international support center, at 1-800-547-8949 (US and Canada) or +1-530-478-4148.

To obtain a local phone number for the support center nearest you, consult the Contact Us section of Grass Valley's website ([www.grassvalley.com](http://www.grassvalley.com)).

An online form for e-mail contact is also available from the website.

## **Corporate Head Office**

Grass Valley  
3499 Douglas-B.-Floreani  
St-Laurent, Quebec H4S 2C6  
Canada  
Telephone: +1 514 333 1772  
Fax: +1 514 333 9828  
[www.grassvalley.com](http://www.grassvalley.com)