Technical Reference
020-102628-03

Mystique Operate
API Commands

CHRISTIE®

NOTICES and SOFTWARE LICENSING AGREEMENT

Copyright and Trademarks

General

Every effort has been made to ensure accuracy, however in some cases changes in the products or availability could occur which may not be reflected in this document. Christie reserves the right to make changes to specifications at any time without notice. Performance specifications are typical, but may vary depending on conditions beyond Christie's control such as maintenance of the product in proper working conditions. Performance specifications are based on information available at the time of printing. Christie makes no warranty of any kind with regard to this material, including, but not limited to, implied warranties of fitness for a particular purpose. Christie will not be liable for errors contained herein or for incidental or consequential damages in connection with the performance or use of this material. Canadian manufacturing facility is ISO 9001 and 14001 certified.

SOFTWARE LICENSING AGREEMENT

Agreement

a. This Software License Agreement (the "Agreement") is a legal agreement between the end user, either an individual or business entity, ("Licensee") and Christie Digital Systems USA, Inc. ("Christie") for the software known commercially as Christie® Mystique that accompanies this Agreement and/or is installed in the server that Licensee has purchased along with related software components, which may include associated media, printed materials and online or electronic documentation (all such software and materials are referred to herein, collectively, as "Software"). If Licensee does not accept the terms of this Agreement, Licensee will return the Software to Christie in accordance with Christie's standard return policy in place from time to time. BY DOWNLOADING, INSTALLING, ACTIVATING OR USING THE SOFTWARE, THE LICENSEE IS AGREEING TO BE BOUND BY THE TERMS OF THIS AGREEMENT.

b. The Software contains third party software programs which require notices and/or additional terms and conditions. Such required third party notices and/or additional terms and conditions are located in the Mystique application (**Help** > **About**). By accepting this Agreement and using the Software, you are also accepting any such additional terms and conditions applicable to the third party software programs. To the extent any license for third party software programs requires an offer to provide source code for those third party software programs, this Agreement constitutes such an offer. You may request more information or a copy of such source code by contacting Christie at open.source@christiedigital.com. In conjunction with such request, provide the Christie Product Number, the Software version and the Product Serial Number (if applicable). If applicable, a nominal cost to cover shipping and media charges will be charged.

License and Restrictions

a. **Ownership of Software**: Christie retains title and ownership of all rights, including copyright and all other intellectual property rights, in and to the Software and all copies thereof, regardless of form or media. This license is not a sale of any rights in or to the Software or any copy thereof. Licensee acquires no right whatsoever to the Software except the right to use the Software in accordance with the terms of this License.

b. **Grant of License**: Christie grants to Licensee a single license providing a nonexclusive and non-transferable right to use, for personal or internal business purposes, the executable version of the Software, as long as Licensee complies with the terms and conditions of this License. Christie reserves all rights not expressly granted to Licensee.

c. **Copy Restrictions**: Where the Software is provided for download onto a personal computer or similar device, Licensee may make as many copies of the Software as is reasonably required for Licensee's own use with the Device. Licensee may also make a backup copy of the Software which may be used only for reinstallation of the Software. Where the software is pre-installed on a server, Licensee may keep one copy of the Software solely for backup or archival purposes. Unauthorized copying or distribution of the Software is expressly prohibited unless agreed to in writing by Christie. Licensee must reproduce and include the copyright notice on any backup copy of the Software. Licensee may not remove or alter any proprietary rights or copyright notice or identification which indicates the ownership of the Software.

d. **Use Restrictions**: Licensee will not (a) modify, adapt, decompile, disassemble, reverse engineer, or otherwise attempt to derive the source code for the Software, or create derivative works based on the Software; (b) assign this Agreement or redistribute, encumber, sell, transfer, lease, or grant a sublicense of the Software or the license granted herein or otherwise transfer rights to the Software to any person or entity except as and when expressly permitted by this Agreement; (c) directly or indirectly, export or transmit the Software, either alone or bundled with other software, to any country to which such export or transmission is restricted by any applicable laws without the appropriate prior written consent of the applicable government entity or department; and (d) publish or otherwise disclose the Software for others to copy. Licensee will make reasonable efforts to prevent any unauthorized use of the Software by third parties and will advise Licensee's employees of the restrictions upon use restrictions herein.

e. **Upgrades and Updates**: While Christie is not required to do so, Christie may provide Licensee with upgrades or updates to the Software, including but not limited to files that are delivered to Licensee via online transmission or otherwise to "patch" or otherwise modify or enhance the software program. This Agreement covers any upgrades or updates to the Software that Licensee may obtain from Christie or one of its authorized vendors, unless the upgrade or update comes with its own license agreement. If Licensee decides not to download or install an upgrade or update provided by Christie, Licensee understands that this could put the Software at risk to serious security threats or cause the Software to be come unstable. Christie may, at its option, cause an upgrade or update to apply automatically.

Maintenance and Support

a. Christie may, in its sole discretion and without any obligation, provide maintenance and support services related to the Software ("Support Services"). Support Services are governed by Christie's policies and programs described in materials available from Christie. Any supplemental software code provided to Licensee as part of the Support Services will be considered part of the Software and subject to the terms and conditions of this Agreement.

Warranty and Limitation of Liability

a. **No Warranties**: The Software is provided "as is" and Christie makes no warranty of any kind with regard to the Software. Christie will not be liable for errors contained in the Software or in connection with the performance or use of the Software. Christie does not warrant that the Software is free from any defects or that it will operate uninterrupted. Christie warrants that the media on which the Software is distributed will be free from defects in materials and workmanship under normal use for a period of ninety (90) days from the date of delivery. Christie will replace any defective media returned to Christie within the ninety (90) day period. The foregoing is Licensee's sole and exclusive remedy for breach of the warranty given pursuant to this Section. Christie does not make any other warranties, either express or implied, including, but not limited to, implied warranties of merchantability, fitness for a particular purpose, title and non-infringement with regard to the Software.

b. **Limitations of Liability**: IN NO EVENT WILL CHRISTIE BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, LOST REVENUES AND LOST DATA ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF CHRISTIE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CHRISTIE SPECIFICALLY DISCLAIMS ANY WARRANTIES OR CONDITIONS, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE AND ACCOMPANYING DOCUMENTATION, IF ANY, PROVIDED HEREUNDER IS PROVIDED "AS IS". CHRISTIE HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

CHRISTIE FURTHER DISCLAIMS ANY AND ALL REPRESENTATIONS RELATING TO THIRD PARTY SOFTWARE WHICH IS EMBEDDED IN THE SOFTWARE. Some jurisdictions do not allow exclusions or limitations of certain conditions or warranties. Only those exclusions or limitations that are lawful in Licensee's jurisdiction will apply to Licensee and, in such instances, Christie's liability will be limited only to the maximum extent permitted by law.

Termination

a. Christie may terminate this Agreement at any time and for any reason. Christie may also terminate this Agreement if the Licensee breaches or is in default of any obligation of Licensee under this Agreement. Upon termination of this Agreement, the Licensee will immediately (i) cease using the Software and (ii) uninstall the Software from any personal computers, servers or other devices (to the extent it is installed thereon). The Licensee's obligation under this provision will survive the termination of this Agreement.

General

a. **Notices**: Unless otherwise set forth in this Agreement, all notices, or other communications hereunder will be deemed to have been duly given when made in writing and delivered in person, by courier or deposited in the mail, postage prepaid, and registered mail, return receipt requested, and addressed to Licensee at the billing address supplied to Christie by Licensee, and addressed to Christie at Christie Digital Systems USA, Inc., c/o Christie Digital Systems Canada, Inc. at 809 Wellington Street North, Kitchener, Ontario, Canada N2G 4Y7, with a copy (which will not constitute notice) to Christie's Legal Department at the same address.

b. **Severability and Waiver**: Any provision of this Agreement which is held by any court to be unenforceable or invalid will be deemed severed from this Agreement, without affecting any other provision of this Agreement. No waiver by Christie of any particular default or omission committed by Licensee will affect or impair the right of Christie in respect of any subsequent default or omission of the same or a different kind. No delay or failure by Christie to exercise any rights in connection with any default or omission committed by Licensee will affect or impair Christie's rights in respect of that particular default or omission or any subsequent default or omission of the same or different kind. In any event, time will continue to be of the essence without the necessity of specific reinstatement.

c. **Governing Law**: This Agreement is governed by the laws of the State of California, without regard to its conflict of law principles. Any suit or action arising out of or in connection with this Agreement will be brought in the federal or state courts located in Orange County, State of California. Each of Licensor and Licensee hereby irrevocably submit to the jurisdiction of such courts for the purpose of such suit or action and expressly and irrevocably waive, to the fullest extent permitted by law, any objection it may now or hereafter have to the venue of any such suit or action in any such court and any such claim that any suit or action has been brought in an inconvenient forum.

Content

Using API commands

This document describes Christie REST and JSON TCP/IP API commands.

The following REST and JSON TCP/IP API commands can be exercised using a variety of tools including cURL and Postman, or by writing your own web client.

Connections to the TCP/IP Operate service can be made on port 3004. Messages, responses, and errors follow the JsonRPC format. For more information on the format, refer to *https://www.jsonrpc.org/specification*.

Product documentation

For installation, setup, and user information, see the product documentation available on the Christie website. Read all instructions before using or servicing this product.

1. Access the documentation from the Christie website:

   - Go to this URL: *http://bit.ly/2lNXz54* or
     https://www.christiedigital.com/en-us/business/products/mystique.

   - Scan the QR code using a QR code reader app on a smartphone or tablet.

2. On the product page, select the edition and switch to the **Downloads** tab.

Related documentation

Additional information on this product is available in the following documents.

- *Mystique User Guide (P/N: 020-001134-XX)*
- *Twist User Guide (P/N: 020-101380-XX)*

Approve

Approves the calibration currently applied to a scene.

Approving a calibration stores the warps and blends in a memory slot of the projector. After approving a calibration, any subsequent calibrations are stored in other memory slots. To restore an approved calibration, use the Restore command.

REST API

Commands

Item	Command	Description
URL	`/api/automation/approve`	—
Method	`POST`	—
Optional URL parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: `200` Content: None	—
Error	Code: `422` Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.

Calibration approval error	Code: `500`
	Content-Type: application/xml
		Displays an error message in XML format, according to the uiredirect specification because a problem occurred when approving a calibration.
		XML error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`    <uiredirect>`
		`        <message type="critical">`
		`            <title>Action Failed</title>`
		`            <text>Current slot id (1)`
		`                cannot be approved; this`
		`                can happen if Mystique`
		`                has not been run at least`
		`                once with this slot.</text>`
		`        </message>`
		`    </uiredirect>`

Example

Approve the last calibration of scene 2:
```
curl -X POST http://localhost/api/automation/approve?scene=2
```

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:approve`	—
Optional parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: `422` Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Message: string	Returns an error object containing a code and message.

Calibration approval error	Code: `500`
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"Current slot id (2) cannot be`
		`        approved; this can happen if Mystique`
		`        has not been run at least once with`
		`        this slot.","code":500}}`

Example

Approve the last calibration of scene 2:

`{"jsonrpc":"2.0","method":"operate:approve","params":{"scene":2},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Related information

Cancel

Cancels any operation that is in progress and is associated with the specified scene.

In some cases, operations such as `Run` might take longer than expected to complete. Use the Cancel command to terminate Christie Mystique processes taking longer than expected, such as:

- Run
- Test Pattern
- Approve
- Restore
- Clear

While multiple operations may be in progress on the sever side, only one operation can be in progress for each scene.

If you cancel a run in progress, test patterns might remain on the projectors without warps and blends. To remove the test patterns, use the Clear command. To restore settings to a previously approved state after canceling a run, use the Restore command or complete a Run.

REST API

Commands

Item	Command	Description
URL	`/api/automation/cancel`	—
Method	`POST`	—
Optional URL parameter	`scene=[0]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	—
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Example

Cancel an operation:
```
curl -X POST http://localhost/api/automation/cancel
```

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:cancel`	
Optional parameter	`scene=[0]`	Specifies the scene by index number. 0 (Default)

Item	Command	Description
Optional parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Example

Cancel an operation:

{"jsonrpc":"2.0","method":"operate:cancel","params":{"scene":0},"id":1}

Result:

{"id":1,"jsonrpc":"2.0", "result":[]}

Related information

Clear

Clears the test pattern for a specified scene.

REST API

Commands

Item	Command	Description
URL	/api/automation/clear	—
Method	POST	—
Optional URL parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional URL parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Invalid request	Code: 422 Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Content-Type: text/plain	Displays a plain text error message.

XML error	Code: 500
	Content-Type: application/xml
		Displays an error message in XML format, according to the uiredirect specification because a problem occurred when clearing a test pattern.
		XML error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<uiredirect>`
		`    <message type="critical">`
		`        <title>Open File Failed</title>`
		`        <text>Unable to open`
		`            'Multi1.cal'. There was an`
		`            error communicating with:`
		`            Camera_01 192.168.6.11`
		`            Please verify the camera`
		`            settings and try again.`
		`            </text>`
		`    </message>`
		`</uiredirect>`

Example

Clear test patterns in scene 0: `curl -X POST http://localhost/api/automation/clear`
Clear test patterns in the scene named First: `curl -X POST http://localhost/api/automation/clear?sceneName=First`
Clear test patterns in scene 2: `curl -X POST http://localhost/api/automation/clear?scene=2`

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:clear`	—
Optional parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)

Item	Command	Description
Optional parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: 422 Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Clearing test pattern error	Code: 500
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`{"message":"Unable to open '1x2.cal'. There`
		`was an error\ncommunicating with:`
		`\n\n\tCamera_01\n\t192.168.1.11\n\n`
		`Please verify the camera settings and`
		`try again.","code":500}}`

Example

Clear a scene:

`{"jsonrpc":"2.0","method":"operate:clear","params":{"scene":0},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Disengage Guardian

Turns off automatic continuous healing that was started using the Engage Guardian command.

REST API

Commands

Item	Command	Description
URL	/api/automation/disengageGuardian	—
Method	POST	—
Optional URL parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional URL parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not

Item	Command	Description
		unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Invalid request	Code: 422 Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Content-Type: text/plain	Displays a plain text error message.

Example

Stop Guardian calibration for scene 0: `curl -X POST http://localhost/api/automation/disengageGuardian`
Stop Guardian calibration for the scene with the name First: `curl -X POST http://localhost/api/automation/disengageGuardian?sceneName=First`
Stop Guardian calibration for scene 2: `curl -X POST http://localhost/api/automation/disengageGuardian?scene=2`

JSON TCP/IP API

Commands

Item	Command	Description
Port	3004	—
Method	operate:disengageGuardian	—
Optional parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: 422 Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500	Returns an error object containing a code and message.

Item	Response	Description
	Message: string	

Example

```
Stop Guardian calibration for scene 0:
{"jsonrpc":"2.0","method":"operate:disengageGuardian","params":{"scene":0},"id":1}
Result:
{"id":1,"jsonrpc":"2.0", "result":[]}
```

Engage Guardian

Turns on continuous healing using Guardian.

Mystique Operate repeatedly performs the Heal operation as long as Guardian protection is engaged. Use the Disengage Guardian command to stop continuous healing.

REST API

Commands

Item	Command	Description
URL	`/api/automation/engageGuardian`	—
Method	`POST`	—
Optional URL parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: `200` Content: None	—
Invalid request	Code: `422` Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.

Example

```
Start Guardian calibration for scene 0:
curl -X POST http://localhost/api/automation/engageGuardian
```

Start Guardian calibration for the scene with the name First:
`curl -X POST http://localhost/api/automation/engageGuardian?sceneName=First`

Start Guardian calibration for scene 2:
`curl -X POST http://localhost/api/automation/engageGuardian?scene=2`

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:engageGuardian`	—
Optional parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: `422` Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Message: string	Returns an error object containing a code and message.

Example

Start Guardian calibration for scene 0:
`{"jsonrpc":"2.0","method":"operate:engageGuardian","params":{"scene":0},"id":1}`
Result:
`{"id":1,"jsonrpc":"2.0", "result":[]}`

Get Settings

Displays the settings for .

Settings are stored in the Settings property of the returned object and responses may return a large amount of data. Do not post the entire response to a Set Settings command.

REST API

Commands

Item	Command	Description
URL	`/api/automation/settings`	—
Method	`GET`	—

Responses

Item	Response	Description
Success	Code: `200` Content type: application/json	Displays configured settings, such as the path to the Christie Mystique .exe file or a list of scene names with associated paths to .cal files.
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.

Example

Get settings for a scene:

`curl http://localhost/api/automation/settings`

Result:

```
{"settings":
    {"exePath": "C:\\Program Files\\Christie Digital Systems\\Mystique 1.2
        \\Mystique.exe", "scenes": [
            {"name": "First",
                "calPath": "C:\\Example\\Multi\\Multi1.cal"},
            {"name": "Second",
                "calPath": "C:\\Example\\Multi\\Multi2.cal"},
            {"name": "Third",
                "calPath": "C:\\Example\\Multi\\Multi3.cal"}]},
    "problems": null}
```

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:getSettings`	—

Responses

Item	Response	Description
Success	Result object	Returns a result object containing configured settings, such as the path to the Christie Mystique .exe file or a list of scene names with associated paths to .cal files.
Generic error	Code: `500`	Returns an error object containing a code and message.

Item	Response	Description
	Message: string	

Example

```
Get settings for a scene:

{"jsonrpc":"2.0","method":"operate:getSettings","params":{},"id":1}

Result:

 {"id": 1, "jsonrpc": "2.0", "result":
     {"settings":
         {"exePath": "C:\\Program Files\\Christie Digital Systems\\Mystique 2.4
             \\Mystique.exe","scenes": [
         {"name": "Mobius",
             "calPath": "C:\\models\\mobius\\mobius.cal"}]},
         "issues":
             {"msg": "The specified path or executable file does not exist.
                 Please enter the full path to the executable.", "error": true}}}
```

Heal

Runs the Guardian calibration for the specified scene.

REST API

Commands

Item	Command	Description
URL	/api/automation/heal	—
Method	POST	—
Optional URL parameter	scene=[integer]	Specifies the scene by index number. 0sceneName=[string] (Default)
Optional URL parameter		Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Invalid request	Code: 422 Content-Type: text/plain	Indicates a sceneName=[stringAn invalid scene index or other invalid data was provided.
Generic error	Code: 500 Content-Type: text/plain	Display a plain text error message.

Item	Response	Description

XML error	Code: 500
	
	Content-Type: application/xml
		Displays an error message in XML format, according to the uiredirect specification because a problem occurred when clearing a test pattern.
		
		XML error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<uiredirect>`
		`    <message type="critical">`
		`        <title>Open File Failed</title>`
		`        <text>Unable to open`
		`            'Multi1.cal'. There was an`
		`            error communicating with:`
		`            Camera_01 192.168.6.11`
		`            Please verify the camera`
		`            settings and try again.`
		`        </text>`
		`    </message>`
		`</uiredirect>`

Example

Run the Guardian calibration for scene 0: `curl -X POST http://localhost/api/automation/heal`
Run the Guardian calibration for the scene with the name First: `curl -X POST http://localhost/api/automation/heal?sceneName=First`
Run the Guardian calibration for scene 2: `curl -X POST http://localhost/api/automation/heal?scene=2`

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:heal`	—
Optional parameter	`scene=[integer]`	Specifies the scene by index number. 0sceneName=[string] (Default)
Optional parameter		Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: `422` Message: string	Indicates an invalid scene index or other invalid data was provided.

Item	Response	Description
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Healing error	Code: 500
	Message: string
		Returns an error objectXML format, according to the uiredirect specification because a problem occurred when clearing a test pattern.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"Cannot perform a`
		`        Guardian Run; feature is not`
		`        licensed.","code":500}}`

Example

Run the Guardian calibration for scene 0:

`{"jsonrpc":"2.0","method":"operate:heal","params":{"scene":0},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Restore the calibration to the last version

Restores the calibration to the last saved calibration.

To approve a calibration, use the Approve command. To restore a specific approved calibration, use the Restore command.

REST API

Commands

Item	Command	Description
URL	`/api/automation/restorelast`	—
Method	`POST`	—
Optional URL parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Error	Code: 422 Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.

Item	Response	Description
Generic error	Code: 500 Content-Type: text/plain	Displays a plain text error message.

Reversion error	Code: 500
	Content-Type: application/xml
		Displays an error message in XML format, according to the uiredirect specification because a problem occurred when restoring a calibration.
		XML error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<uiredirect>`
		`    <message type="critical">`
		`        <title>Open File Failed</title>`
		`        <text>Unable to open`
		`            'Multi1.cal'. There was`
		`            an error communicating`
		`            with: Camera_`
		`            01 192.168.6.11`
		`            Please verify the camera`
		`            settings and try again.`
		`        </text>`
		`    </message>`
		`</uiredirect>`

Example

Restore the last approved calibration:
```
curl -X POST http://localhost/api/automation/restorelast
```

JSON TCP/IP API

Commands

Item	Command	Description
Post	3004	—
Method	operate:restoreLast	—
Optional parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: 422 Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500	Returns an error object containing a code and message.

Item	Response	Description
	Message: string	

Restore error	Code: `500`
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"Unable to open '1x2.cal'. There`
		`        was an error\ncommunicating with:`
		`        \n\n\tCamera_01\n\t192.168.1.11\n\n`
		`        Please verify the camera settings and`
		`        try again.","code":500}}`

Example

Restore the last approved calibration:

`"jsonrpc":"2.0","method":"operate:restoreLast","params":{"scene":0},"id":1`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Related information

Restore the calibration to a specific version

Restores the current calibration to a previously approved calibration.

To obtain a list of approved calibrations, use the Scenes command. To approve a calibration, use the Approve command.

REST API

Commands

Item	Command	Description
URL	`/api/automation/restore/:id`	—
Method	`POST`	—
Required URL parameter	`:id=[integer]`	Provides the ID of the intended calibration to restore. To obtain an ID for an approved calibration, use the Scenes command.
Optional URL parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Error	Code: 422 Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Content-Type: text/plain	Displays a plain text error message.
Reversion error	Code: 500 Content-Type: application/xml	Displays an error message in XML format, according to the uiredirect specification because a problem occurred when restoring a calibration. XML error: ```xml <?xml version="1.0" encoding="UTF-8"?> <uiredirect> <message type="critical"> <title>Open File Failed</title> <text>Unable to open 'Multi1.cal'. There was an error communicating with: Camera_01 192.168.6.11 Please verify the camera settings and try again. </text> </message>```

Example

Restore scene 5 to the calibration with ID 2:
```
curl -X POST http://localhost/api/automation/restore/2?scene=5
```

JSON TCP/IP API

Commands

Item	Command	Description
Port	3004	—
Method	operate:restore	—
Required parameter	slotId=[integer]	Provides the ID of the intended calibration to restore. To obtain an ID for an approved calibration, use the Scenes command.
Optional parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional parameter	sceneName=[string]	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: `422` Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Message: string	Returns an error object containing a code and message.

Reversion error	Code: `500`
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"Slot id (3) cannot be restored;`
		`        it is either invalid or has not`
		`        previously been approved.`
		`        ","code":500}}`

Example

Restore scene 0 to the calibration with ID 1:

`{"jsonrpc":"2.0","method":"operate:restore","params":{"slotId":1, "scene":0},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Related information

Run

Calibrates the specified scene.

The response is not returned until calibration is complete. Closing a connection when making a request does not stop the process and the calibration continues on the server.

To stop an operation that is in progress, use the Cancel command. If canceling a run in progress, test patterns might remain on the projectors without warps and blends. To remove the test patterns, use the Clear command. To restore settings to a previously approved state after canceling a run, use the Restore command or complete a Run.

REST API

Commands

Item	Command	Description
URL	`/api/automation/run`	—
Method	`POST`	—

Item	Command	Description
Optional URL parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional URL parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Code: `200` Content: None	—
Error	Code: `422` Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.

Calibration error	Code: `500`
	Content-Type: application/xml
		Displays an error message in XML format, according to the uiredirect specification because a problem occurred when running a calibration.
		XML error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<uiredirect>`
		`    <message type="critical">`
		`        <title>Open File Failed</title>`
		`        <text>Unable to open 'Multi1.cal'.`
		`            There was an error`
		`            communicating with:`
		`            Camera_01 192.168.6.11`
		`            Please verify the camera`
		`            settings and try again.`
		`            </text>`
		`    </message>`
		`</uiredirect>`

Example

Calibrate scene 2: `curl -X POST http://localhost/api/automation/run?scene=2`
Calibrate the scene with the name First: `curl -X POST http://localhost/api/automation/run?sceneName=First`
Calibrate scene 0: `curl -X POST http://localhost/api/automation/run`

JSON TCP/IP API

Commands

Item	Command	Description
Port	`3004`	—
Method	`operate:run`	—
Optional parameter	`scene=[integer]`	Specifies the scene by index number. 0 (Default)
Optional parameter	`sceneName=[string]`	Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: 422 Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Restore error	Code: 500
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"The projectors must be powered`
		`        on and unshuttered before proceeding.`
		`        \nEnsure the light path is clear of`
		`        people.\n\nIs it safe for`
		`        the projectors to emit light?",`
		`        "code":500}}`

Example

Calibrate scene 0:

`{"jsonrpc":"2.0","method":"operate:run","params":{"scene":0},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Related information

Cancel (on page 7)
Clear (on page 9)
Restore the calibration to a specific version (on page 20)
Restore the calibration to the last version (on page 18)

Scenes

Displays a list of scenes.

REST API

Commands

Item	Command	Description
URL	`/api/automation/scenes`	—
Method	GET	—

Responses

Item	Response	Description
Success	Code: `200` Content-Type: `application/json` Content: An object providing the name, calPath, and approved slots for each scene	—
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.

Example

Obtain a list of scenes:

```
curl http://localhost/api/automation/scenes
```

Response:

```
{"scenes": [
   {"name": "First",
     "calPath": "C:\\Example\\Multi\\Multi1.cal"},
   {"name": "Second",
     "calPath": "C:\\Example\\Multi\\Multi2.cal"},
   {"name": "Third",
     "calPath": "C:\\Example\\Multi\\Multi3.cal"}],
   "sceneCache":
     {"0":
       {"info":
         {"TimeOfLastRun":
           [""],
         "LastRunDiagFolder":
           [""],
         "CurrentSlotID":
           ["2"],
         "Slot": [
             {"$":
               {"ID": "3"},
             "Name":
               ["2016-07-18 10:38 AM"]},
           {"$":
               {"ID": "4"},
             "Name":
               ["2017-07-11 8:56 AM"]},
```

```
            {"$":
                {"ID": "1"},
              "Name":
                ["2017-07-11 1:08 PM"]}]}},
        "1":
        {"info":
            {"TimeOfLastRun":
                [""],
              "LastRunDiagFolder":
                [""],
              "CurrentSlotID":
                ["1"],
              "Slot": [
                {"$":
                    {"ID": "3"},
                  "Name":
                    ["2016-07-22 3:00 PM"]},
                {"$":
                    {"ID": "4"},
                  "Name":
                    ["2017-07-11 10:08 AM"]}]}}},
        "2":
        {"info":
            {"TimeOfLastRun":
                ["2017-07-11 10:08 AM"],
              "LastRunDiagFolder":
                ["C:\\ProgramData\\Christie Digital Systems\\Mystique 1.2
                \\Diagnostics\\2017-07-11_10-08-25"], "CurrentSlotID":
                ["3"]}}}}
```

Each element of this array is an object representing the approved calibration. The array of approved calibrations is stored in `sceneCache[sceneIndex].info.Slot`. The object keys for the `sceneCache` are scene index values. For example, if referring to this object as `slot`, it has `slot.$.ID` and `slot.Name`. If restoring a previous calibration, the `ID` value is passed to the Restore command.

JSON TCP/IP API

Commands

Item	Command	Description
Port	3004	—
Method	operate:listScenes	—

Responses

Item	Response	Description
Success	Result	Returns a result object providing the name, calPath, and approved slots for each scene.
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Example

Obtain a list of scenes:

```
{"jsonrpc":"2.0","method":"operate:listScenes","params":{},"id":1}
```

Result:

```
{"id":1,
"jsonrpc":"2.0",
"result":
  {"scenes": [
     {"name": "First",
       "calPath": "C:\\Example\\Multi\\Multi1.cal"},
     {"name": "Second",
       "calPath": "C:\\Example\\Multi\\Multi2.cal"},
     {"name": "Third",
       "calPath": "C:\\Example\\Multi\\Multi3.cal"}],
   "sceneCache":
     {"0":
       {"info":
         {"TimeOfLastRun":
           [""],
          "LastRunDiagFolder":
           [""],
          "CurrentSlotID":
           ["2"],
          "Slot": [
            {"$":
               {"ID": "3"},
              "Name":
                ["2016-07-18 10:38 AM"]},
             {"$":
               {"ID": "4"},
              "Name":
                ["2017-07-11 8:56 AM"]},
             {"$":
               {"ID": "1"},
              "Name":
                ["2017-07-11 1:08 PM"]}]}},
      "1":
       {"info":
         {"TimeOfLastRun":
           [""],
          "LastRunDiagFolder":
           [""],
          "CurrentSlotID":
           ["1"],
          "Slot": [{
             "$":
               {"ID": "3"},
              "Name":
               ["2016-07-22 3:00 PM"]},
             {"$":
               {"ID": "4"},
              "Name":
                ["2017-07-11 10:08 AM"]}]}},
      "2":
       {"info":
         {"TimeOfLastRun":
           ["2017-07-11 10:08 AM"],
          "LastRunDiagFolder":
           ["C:\\ProgramData\\Christie Digital Systems\\Mystique Install 1.2
           \\Diagnostics\\2017-07-11_10-08-25"],
          "CurrentSlotID":
           ["3"]}}}}}
```

Set Settings

Defines the scene settings. A settings object is sent that contains an array of scene objects.

REST API

Commands

Item	Command	Description
URL	`/api/automation/settings`	—
Method	`POST`	—
Data parameters	Content-Type: `application/json`	A settings object includes the following properties: • **exePath**—Full path to the Christie Mystique executable on the server • **scenes**—Array of scene objects A scene object includes the following properties: • **name**—Human-readable name of the scene • **calPath**—Full path to the .cal file for the scene

Responses

Item	Response	Description
Success	Code: `200` Content: None	—
Generic error	Code: `500` Content-Type: text/plain	Displays a plain text error message.
Failed validation	Code: `500` Content-Type: `application/json`	Receives a response body if a validation error occurs. The text of the error message appears in the `msg` property. Validation error: `{"msg": "The specified executable file does not exist."}`

Example

Write the settings.json file to disk, with these settings:

```
{"exePath": "C:\\CustomInstallLocation\\Mystique.exe", "scenes": [
    {"name": "First",
        "calPath": "C:\\Example\\Multi\\Multi1.cal"},
      {"name": "Second",
        "calPath": "C:\\Example\\Multi\\Multi2.cal"},
```

```
        {"name": "Third",
          "calPath": "C:\\Example\\Multi\\Multi3.cal"}]]
```

Command:

```
curl -v POST http://localhost/api/automation/settings -d @settings.json --header
"Content-Type: application/json"
```

JSON TCP/IP API

Commands

Item	Command	Description
Port	3004	—
Method	operate:setSettings	—
Required parameters	Settings object: exePath=[string] scenes = [array of Scene objects] Scene object: name=[string] calPath=[string]	A settings object includes the following properties: • **exePath**—Full path to the Christie Mystique executable on the server • **scenes**—Array of scene objects A scene object includes the following properties: • **name**—Human-readable name of the scene • **calPath**—Full path to the .cal file for the scene

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.

Failed validation	Result	Returns a result object specifying the failed validation issues.
		
		`{"id":1,"jsonrpc":"2.0","result":`
		`    {"issues":`
		`        {"msg":"The specified path or`
		`            executable file does not`
		`            exist. Please enter the`
		`            full path to the`
		`            executable."}}}`

Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Example

Write the settings.json file to disk, with these settings:

```
{"jsonrpc":"2.0","method":"operate:setSettings","params": {
    "exePath": "C:\\Program Files\\Christie Digital Systems\\Mystique 2.4\
\Mystique.exe",
```

```
    "scenes": [
        {"name": "Mobius",
            "calPath": "C:\\models\\mobius\\mobius.cal"}]},
    "id":1}
```

Result:

```
{"id":1,"jsonrpc":"2.0", "result":[]}
```

Error:

```
 {"id": 1,
     "jsonrpc": "2.0",
     "result":
         {"issues":
             {"msg": "The specified path or executable file does not exist.
                 Please enter the full path to the executable.", "error": true}}}
```

Test Pattern

Displays a test pattern on a projector.

REST API

Commands

Item	Command	Description
URL	/api/automation/testpattern	—
Method	POST	—
Optional URL parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional URL parameter		Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.sceneName=[string]

Responses

Item	Response	Description
Success	Code: 200 Content: None	—
Error	Code: 422 Content-Type: text/plain	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Content-Type: text/plain	Displays a plain text error message.
Test pattern error	Code: 500 Content-Type: application/xml	Displays an error message in XML format, according to the uiredirect specification because a

Item	Response	Description

		sceneName=[stringA problem occurred when displaying a test pattern.
		
		Test pattern error:
		
		`<?xml version="1.0" encoding="UTF-8"?>`
		`    <uiredirect>`
		`        <message type="critical">`
		`            <title>Open File Failed`
		`                </title>`
		`            <text>Unable to open`
		`                'Multi1.cal'.`
		`                There was an error`
		`                communicating with:`
		`                Camera_01 192.168.6.11`
		`                Please verify the`
		`                camera settings`
		`                and try again.`
		`                </text>`
		`        </message>`
		`    </uiredirect>`

Example

Display a test pattern:
`curl -X POST http://localhost/api/automation/testpattern`

JSON TCP/IP API

Commands

Item	Command	Description
Port	3004	—
Method	operate:testPattern	—
Optional parameter	scene=[integer]	Specifies the scene by index number. 0 (Default)
Optional parameter		Specifies the scene by exact name. The name is case-sensitive. If the names are not unique, the first matching (lowest index) scene is returned.sceneName=[string]

Responses

Item	Response	Description
Success	Result	Indeterminant value and may change in future.
Invalid request	Code: 422 Message: string	Indicates an invalid scene index or other invalid data was provided.
Generic error	Code: 500 Message: string	Returns an error object containing a code and message.

Item	Response	Description

Test pattern error	Code: 500
	
	Message: string
		Returns an error object containing a code and message.
		
		`{"id":1,"jsonrpc":"2.0","error":`
		`    {"message":"Unable to open '1x2.cal'.`
		`        There was an error\ncommunicating with:`
		`        \n\n\tCamera_01\n\t192.168.1.11\n\n`
		`        Please verify the camera settings and`
		`        try again.",`
		`        "code":500}}`

Example

Display a test pattern:

`{"jsonrpc":"2.0","method":"operate:testPattern","params":{"scene":0},"id":1}`

Result:

`{"id":1,"jsonrpc":"2.0", "result":[]}`

Corporate offices

Christie Digital Systems USA, Inc.
ph: 714 236 8610

Christie Digital Systems Canada Inc.
ph: 519 744 8005

Worldwide offices

Africa
ph: +27 (0)11 510 0094

Australia
ph: +61 (0) 7 3624 4888

Brazil
ph: +55 (11) 2548 4753

China (Beijing)
ph: +86 10 6561 0240

China (Shanghai)
ph: +86 21 6030 0500

Columbia
ph: +57 (318) 447 3179

Germany
ph: +49 (0) 221 99512 0

India
ph: +91 (080) 6708 9999

Japan (Tokyo)
ph: 81 3 3599 7481

Korea (Seoul)
ph: +82 2 702 1601

Mexico
ph: +52 55 4744 1790

Singapore
ph: +65 6877 8737

Spain
ph: +34 91 633 9990

Middle East
ph: +971 (0) 503 6800

United Kingdom
ph: +44 (0) 118 977 8000

United States (Arizona)
ph: 602 943 5700

Independent sales consultant offices

Italy
ph: +39 (0) 2 9902 1161

Russia
ph: +36 (0) 1 47 48 100

For the most current technical documentation, visit www.christiedigital.com.