STARKEEPER.IT

Voyager RoboTarget JSON-RPC API

Definitions, Events and Methods under NDA

Leonardo Orazi

June 25, 2024

INDEX

1. Introduction

VOYAGER have an internal Application Server that allow external application to interact with it :

- receiving events
    - setup events
    - action events
    - error events
- send commands
    - setup cmd
    - action run
    - profile management
    - environment manage

This document are reserved to the RoboTarget JSON-RPC API . For workflow and connection management please refer to the Application Server Documentation. The API about RoboTarget working only in a Voyager installation having the Advanced/Full license with RoboTarget NDA Plug in activated.

What are inside this document are totally under NDA , is not possible disclose it to public or other entities outside of what/who are reported in the NDA you have signed.

2. RoboTarget Data Objects

Profile	All the settings that belong to a complete astrophotography setup configured in Voyager. Match to the Voyager Profile.
Base Sequence	Sequence file to use like template for inherits the final Sequence to use for shot the Target
Set	Group the target for some kind of logic, or kind of target, or period and so and so. If the Set is disabled all the Target inside will not be considered by the RoboTarget Scheduler
Target	Deep sky object to shot with all information about constraints, pointing ad so.If the Disabled Targets are not used in Scheduler process
Shot	Shot information by filter with exposure and shot settings. Disabled Shots are not used in Sequence creation
Session	Set of Run done for the specified target
Run	Sequence runned on a Target
Orphan Set	Set that not belong to any profile (deleted or not exists). Target inside a Orphan Set cannot be used by Scheduler
Orphan Target	Target that have a Base Sequence not exists (deleted or belong to another profile) . Orphan Target cannot be used by Scheduler

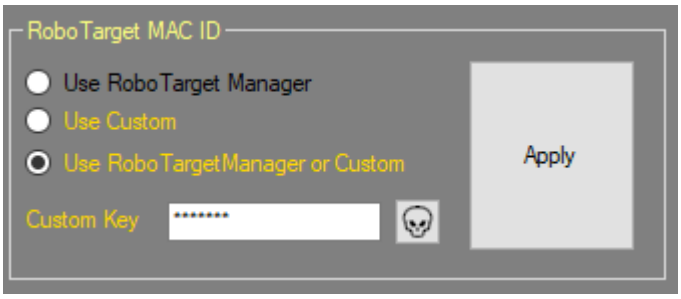
3. Setup the MAC key related to NDA in Voyager Installation/Setup

With purchase of License and sign of the NDA you have access to reserved API of Voyager Application Server dedicated to RoboTarget (API available only for Voyager License Advanced or Full).

To access to this reserved API you need a MAC hashing with dedicated key and words, this key and words (strings) are released to you with the License.

This is the operation to do for configure Voyager installation to be able to use your MAC key and words:

- Install the version of Voyager recommended to you by the Voyager Developer
- Open Voyager
- Go to Setup Section on general menù
- Open the Common Tab
- Locate the RoboTarget MAC box



- Select “Use Custom” or “Use RoboTargetManager or Custom”
- Edit the Custom Key with the MAC key you have received
- Press Apply
- Restart Voyager

About The MACID mode (useful to lock at higher level the access to RoboTarget API):

- “Use RoboTarget Manager” is the default for NOT NDA Server. This allow only the Legacy Voyager RoboTarget Manager Application to connect to the RoboTarget API.
- “Use Custom” is reserved to NDA Server and allow access to RoboTarget Api only using the Custom Key and MAC system. Voyager RoboTarget Manager cannot access to the Server
- “User RoboTarget Manager or Custom” allow access to RoboTarget API from Voyager RoboTarget Manager and from Custom KEY and MAC system

4. Activate for First the RoboTarget Client Mode to Use this API

At each connection with the Voyager Application Server remember to set the RoboTarget mode in your client for first after the user authentication. To do this use the command `RemoteSetRoboTargetManagerMode` (always wait before for the Event Version).

Example of client connection JSON-RPC trace:

(RX) => {“Event”:”Version”,”Timestamp”:1652231344.88438,”Host”:”RC16”,”Inst”:1,”VOYVersion”:”Release 2.3.5s – Built 2022-05-08”,”VOYSubver”:””,”MsgVersion”:1}

(TX) => {“method”: “AuthenticateUserBase”, “params”: {“UID”:”666d6e78-6130-417c-b896-d03c8c28208c”,”Base”:”OTdDRjZFRjM3Mul9KFMELKEFjBBRDY2MjYxMzJDRkUzOTFEMkY4MjU1Rjc3OERDOTcyMkIyODk3QzBFMkI1MzEwOUJCNUI2REY2OEFEQk0EFJIIEFNGMUNEQ0E2MTBDQ0ZENzJBNUJBMzQxNDQ1OA==”}, “id”: 2}

(TX) => {“method”: “RemoteSetRoboTargetManagerMode”, “params”: {“UID”:”4ba87f87-b3e8-4bf5-9314-287ebc1c70f7”,”MACKey”:”xxxxxx”,”Hash”:”mQw/4x7qn09944Ndj5ne9/Z+b0=”}, “id”: 3}

(RX) => {“Event”:”RemoteActionResult”,”Timestamp”:1652391345.33749,”Host”:”RC16”,”Inst”:1,”UID”:”4ba87f87-b3e8-4bf5-9314-287ebc1c70f7”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“ret”:”DONE”, “VersionDB”:21}}

5. RoboTarget Events

a) RoboTargetRunningTargetEphemerisPNG

If the client is declared as RoboTarget Manager, at beginning of each target sequence during the scheduler Voyager will send a PNG of the ephemeris of the target like the one showed in the RoboTarget section form. Image sent in base64 format.

Attribute	Type	Description
TargetUID	String	UID of the Target
TargetTAG	Integer	TAG of the Target
TargetName	String	Target Name
Base64Data	String	PNG image data in Base64 format
IsValid	boolean	True if the image and the data about have sense, otherwise you will found a blank image with a yellow text for the reason
MinutesForPixel	double	How many minutes match 1 pixel in the X axis
StartDateTime	datetime	Define which is the datetime associated to the first pixel on the left of the x axis, expressed in local time
StartDateTimeUTC	datetime	Define which is the datetime associated to the first pixel on the left of the x axis, expressed in UTC time
EndDateTime	datetime	Define which is the datetime associated to the last pixel on the right of the x axis, expressed in local time
EndDateTimeUTC	datetime	Define which is the datetime associated to the last pixel on the right of the x axis, expressed in UTC time

You can decide to draw directly on the image a vertical line to show the actual datetime or other meanings calcultating the difference in minutes between the datetime interested and the StartDateTime and using the MinutesForPixel field calculate the offset in pixel to add from the position 0 in the x axis to draw the line in a desired datetime position.

Example:

{"Event":"RoboTargetRunningTargetEphemerisPNG","Timestamp":1678140994.07035,"Host":"ORIONE","Inst":1,"TargetUID":"8b271e8b-76c6-4b6c-a4eb-dc7a810d052d","TargetTAG":"","TargetName":"12950-0-NGC1952_R","Base64Data":"
/1dW1AIHv75w7c1/mvTfvJUFwKe+Pr2fm3ntmJuR83jn3zn1RiB4CLrnk0h8U/6fB2LH5U5hJG6l///5SPYI7IWzIROyvswKZVbYgvUIC0p+LQ3p5shXIlmMbjwxpE3Co1FocKrkeh0utoeN1OFJyDY6QPVxxHZJEpFSiiCDxcRRpPh2zNel7xknfvfYr5qTPkd8zTvoc+R……."}

6. RoboTarget Commands

VOYAGER provides an RPC (remote procedure call) interface for clients. The message protocol is JSON RPC 2.0.

Requests are sent as a single line of text, terminated by CR LF. Responses from the server are also a single line of text terminated by CR LF. Pamaters name and parameters value are case sensitive, please for Boolean value use ***true*** or ***false*** lower case.

All the commands (exceptions you'll find in a single command description) return an **async** jsonrpc result or jsonrpc error. You can refer to jsonrpc protocol or see the example below. Remember that ID is a integer counter sequential of the command in the client scope.

All the commands (exceptions you'll find in a single command description) return **when finished** an `RemoteActionResult` event.

All Command (exceptions you'll find in a single command description) have like params a string unique identifier UID, usually used is a windows guide identifier https://en.wikipedia.org/wiki/Universally_unique_identifier . You can use anyway a unique string generated with your rule. This string must identify univoque the command.

Some commands can generate dedicated signal events before to send the `RemoteActionResult` final event.

For more info about command please refer to the Application Server documentation.

a) RemoteSetRoboTargetManagerMode

<table>
<tr><td>Method</td><td colspan="3">RemoteSetRoboTargetManagerMode</td></tr>
<tr><td>Description</td><td colspan="3">Declare to the Server to considering this Client like a RoboTarget Manager. This command must be used for first after user authentication to allow use of all the others API included in this document</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>MACKey</td><td>String</td><td>The MAC Key string received with the NDA</td></tr>
<tr><td>Hash</td><td>String</td><td>Create a concatenated string with "||:||" string separator of RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is) + the 4 MAC strings in order (1 to 4) . Finally make an SHA1 hash and convert to base 64 string, see the example below.</td></tr>
</table></td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full with NDA</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3"><table>
<tr><td>ret</td><td>String</td><td>DONE in case of success o ERROR : <description> in case of error</td></tr>
<tr><td>VersionDB</td><td>Numeric</td><td>Version of remote RoboTarget DB</td></tr>
</table></td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final Hash are not correct

➔ {"method": "RemoteSetRoboTargetManagerMode", "params": {"UID":"4ba87f87-b3e8-4bf5-9314-287ebc1c70f7","MACKey":"xxxxxx","Hash":"mQw/4x7qn09944Ndj5ne9/Z+b0="}, "id": 3}

←{“jsonrpc”: “2.0”, “result”: 0, “id”: 3}

←{“Event”:”RemoteActionResult”,”Timestamp”:1652391345.33749,”Host”:”RC16”,”Inst”:1,”UID”:”4ba87f87-b3e8-4bf5-9314-287ebc1c70f7”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“ret”:”DONE”, “VersionDB”:21}}

Hash creation for this call with this source data (VALID ONLY FOR THIS COMMAND):

RoboTarget Shared Secret = “pippo”

SessionKey= “1652231344.88438”

MAC 1 = “12345678”

MAC 2 = ”abcdefg”

MAC 3 = ”pluto”

MAC4 = “paperino”

Actuating the processing …

String concatenated = “pippo||:||1652231344.88438||:||12345678abcdefgplutopaperino”

SHA1 hashing = “69efafc940cabd1797da7dc57a1452cdaae6d0ff”

After Base64 conversion:

Hash=”NjllZmFmYzk0MGNhYmQxNzk3ZGE3ZGM1N2ExNDUyY2RhYWU2ZDBmZg==”

You can check also with online tools for SHA1 hashing e Base64 conversion:

http://www.sha1-online.com/

https://www.base64encode.org/

b) RemoteRoboTargetGetSet

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetSet</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Set for the Profile defined in Profile Parameter from Voyager ordered by Set Name</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>ProfileName</td><td>String</td><td>Profile name used for search about Set. If empty will be answered all the set for all profile configured in Voyager</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example below.</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>Array</td><td>Array of Set Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>setname</td><td>string</td><td>Name of Set</td></tr>
<tr><td>profilename</td><td>string</td><td>Profile to which the Set Belong</td></tr>
<tr><td>Isdefault</td><td>Boolean</td><td>True if the Set is the Default container for the Profile</td></tr>
<tr><td>status</td><td>integer</td><td>0= Enabled
1= Disabled</td></tr>
<tr><td>tag</td><td>string</td><td>Tag of Set</td></tr>
<tr><td>note</td><td>string</td><td>Text Note if defined</td></tr>
</table></td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetSet”, “params”: {“ProfileName”:””,”UID”:”0697f2f9-24e4-4850-84e9-18ea28b05fe9”,”MAC”:”nWq/V98Laq+hFFdMvynnneAyKvk=”}, “id”: 5}

🡐{“jsonrpc”: “2.0”, “result”: 0, “id”: 5}

🡐{“Event”:”RemoteActionResult”,”Timestamp”:1652224143.95634,”Host”:”ORIONE”,”Inst”:1,”UID”:”0697f2f9-24e4-4850-84e9-18ea28b05fe9”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”f5421b76-40e4-4f06-82fb-b10f76b49295”, “setname”:”Finished”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:1, “tag”:””, “note”:”” },{ “guid”:”161c7e76-d428-40f6-9fd1-f89bdc73c428”, “setname”:”Galaxy”, “profilename”:”ColorTestAdvanced.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”6ce88b57-cebd-4c6a-85f8-bc27ebfd8365”, “setname”:”Narrow HAOIII”, “profilename”:”ColorTestAdvanced.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”fdcb842f-50e8-47d1-bfda-b8d90a08c08d”, “setname”:”Out of Season”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”3861b277-28ac-4894-81fa-7dfcfabe6ddc”, “setname”:”Parked”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:1, “note”:”” },{ “guid”:”5482d20e-2304-41d1-8d2b-32adc2c314bc”, “setname”:”Set Prova”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”18ab233e-0180-4e0b-a513-a87ea6886ba2”, “setname”:”Template”, “profilename”:”Default.v2y”, “isdefault”:false,”status”:1, “note”:”” },{ “guid”:”e73d3bc4-9e13-49f0-b9b9-89fbf03a1a30”, “setname”:”Test Set”, “profilename”:”ColorTestAdvanced.v2y”, “isdefault”:false,”status”:0, “note”:”” }]}}

String to use for Hashing,B64 conversion and MAC creation in this example for RoboTarget secret=”pippo”:

“pippo||😐|1652231344.88438||😐|5||😐|0697f2f9-24e4-4850-84e9-18ea28b05fe9”

c) RemoteRoboTargetGetBaseSequence

<table>
<tr><td>Method</td><td colspan="4">RemoteRoboTargetGetBaseSequence</td></tr>
<tr><td>Description</td><td colspan="4">Return list of RoboTarget Base Sequence for the Profile defined in Profile Parameter from Voyager ordered by Base Sequence Name</td></tr>
<tr><td>Params</td><td colspan="4">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>ProfileName</td><td>String</td><td>Profile name used for search about Set. If empty will be answered all the Base Sequence for all profile configured in Voyager</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="4">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="4">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>Array</td><td colspan="2">Array of Base Sequence Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>basesequencename</td><td>string</td><td>Name of Base Sequence</td></tr>
<tr><td>filename</td><td>string</td><td>Name of Base Sequence file</td></tr>
<tr><td>profilename</td><td>string</td><td>Named of profile name which Base Sequence belong</td></tr>
<tr><td>isdefault</td><td>Boolean</td><td>True if the Set is the Default container for the Profile</td></tr>
<tr><td>Status</td><td>integer</td><td>0= Enabled
1= Disabled</td></tr>
<tr><td>note</td><td>string</td><td>Text Note if defined</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetBaseSequence”, “params”: {“ProfileName”:””,”UID”:”81c4ed08-5062-4dd7-b561-21e9e1bdb90c”,”MAC”:”q3DHb62YtMt/EzWp98qNIu4+QBs=”}, “id”: 6}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 6}

{“Event”:”RemoteActionResult”,”Timestamp”:1652224144.25321,”Host”:”ORIONE”,”Inst”:1,”UID”:”81c4ed08-5062-4dd7-b561-21e9e1bdb90c”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”ae4df8c6-41ca-4°3e-bdf5-594bbab7881a”, “basesequencename”:”BubbleNebula_LRGB.s2q”, “filename”:”BubbleNebula_LRGB.s2q”, “profilename”:”Default.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”19e08806-7734-487d-862e-9cbfdb161779”, “basesequencename”:”ConeNebulaHAO3.s2q”, “filename”:”ConeNebulaHAO3.s2q”, “profilename”:”Default.v2y”, “isdefault”:true,”status”:0, “note”:”” },{ “guid”:”55b76f37-ad12-4°93-981°-03253ba46e22”, “basesequencename”:”DefaultLRGB.s2q”, “filename”:”DefaultLRGB.s2q”, “profilename”:”ColorTestAdvanced.v2y”, “isdefault”:true,”status”:0, “note”:”” },{ “guid”:”24e4fcd5-a14d-41b3-bbc4-655c401b05a4”, “basesequencename”:”SequenzaBase_TestFlatNoMount.s2q”, “filename”:”SequenzaBase_TestFlatNoMount.s2q”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:true,”status”:0, “note”:”” },{ “guid”:”d80a07f9-c56e-4°42-bbc9-58°84c9a3438”, “basesequencename”:”TestRotatoreMeridiano.s2q”, “filename”:”TestRotatoreMeridiano.s2q”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:0, “note”:”” },{ “guid”:”90ae5721-a248-4159-ad74-56e13cf26141”, “basesequencename”:”TestUnguidedNoPlateSolve.s2q”, “filename”:”TestUnguidedNoPlateSolve.s2q”, “profilename”:”TestFlatNoMount.v2y”, “isdefault”:false,”status”:0, “note”:”” }]}}

d) RemoteRoboTargetGetTarget

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetTarget</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Target for the Set defined in RefGuid Parameter from Voyager ordered by Base Target Name</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidSet</td><td>String</td><td>UID of Set which Target Belong. If empty will be return all the Target</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table></td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3"><table>
<tr><td>List</td><td>Array</td><td>Array of Target Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>targetname</td><td>string</td><td>Name of Target</td></tr>
<tr><td>tag</td><td>string</td><td>Tag of Target</td></tr>
<tr><td>refguidset</td><td>string</td><td>UID of Set which Target belong</td></tr>
<tr><td>refguidbasesequence</td><td>string</td><td>UID of Base Sequence</td></tr>
</table></td></tr>
</table></td></tr>
</table>

		which Target belong
raj2000	numeric	RA coordinate of Target in J2000 format expressed in Hours
decj2000	numeric	DEC coordinate of Target in J2000 format expressed in Degree
pa	numeric	Position Angle to use for pointing object (Mechanical PA or Sky PA depends on the base sequence configuration)
datecreation	datetime	Date of target creation
status	integer	Status of target 0=Enabled 1=Disabled
statusop	integer	Operative status of Target -1=Unknow 0=Idle 1=Running 2=Finished 3=Ephemeris not calculated 4=Expired
priority	integer	0=Very Low 1=Low 2=Normal 3=High 4=First
note	string	Text note if defined
isrepeat	boolean	True if all the shot configured will be repeated more times like group in the sequence
repeat	integer	Number of repeat for the previous flag
isfinishactualexposure	boolean	True if finish actual exposure in case of time end expired without abort it
iscoolsetpoint	boolean	Override in the sequence the cooling set point temperature if already enabled the cooling in the base sequence (not switch on cooling)
coolsetpoint	integer	Cooling temperature
iswaitshot	boolean	Override in the sequence the wait time between shot if already enabled in base sequence
waitshot	integer	Time in seconds to wait

		between shot
isguidetime	boolean	Override in the sequence the guiding time exposure if already enabled in base sequence
guidetime	numeric	Exposure time seconds for guiding shot
setname	string	Name of Set which target belong
settag	string	Tag of Set
profilename	string	Name of Profile which target belong
sequencename	string	Name of Base Sequence will be used for take shots
cid	string	UID used only by Voyager RoboTarget Manager
cmask	string	Mask used to understand which constraints are used for this target (see dedicated list below)
caltmin	numeric	Altitude min in degree
csqmmin	numeric	Min SQM
chastart	numeric	HA start in hour
chaend	Numeric	HA end in hour
cdatestart	Datetime	Date Start
cdateend	Datetime	Date End
ctimestart	Datetime	Time Start
ctimeend	Datetime	Time End
cmoondown	boolean	True if you want to check the moon down
cmoonphasemin	integer	Min Moon Phase percentage
cmoonphasemax	integer	Max Moon Phase percentage
cmoondistance	numeric	Distance of moon from Target in Degree
chfdmeanlimit	numeric	Max value of single shot HFD in pixel
cmaxtimeforday	numeric	Minutes max for a target in a single day
cairmassmin	numeric	Air Mass min
cairmassmax	numeric	Air Mass max
cmoondistancelorentzian	Integer	Lorentzian avoidance profile,see table below
cmaxtime	Integer	Minutes max for duration of a sequence for the target
cosdatestart	datetime	Date for oneshot target
costimestart	datetime	Time for oneshot target
cosearly	Integer	Minutes for early start a oneshot target

cpintearly	Integer	Minutes for early start a preset time interval target
cpintreset	boolean	Reset Progress at each sequence run
cpintintervals	Json object	Array of interval in JSON format. See structure of Array in paragraph 7
cmask2	string	Mask used to understand which additional constraints are used for this target (see dedicated list below)
cL01	Boolean	True if apply
cM01	Boolean	True if apply
cN01	Boolean	True if apply
cS01	Boolean	True if apply
auxsessioncontainer	object	Progress is a numeric and report the percentage finished for target
token	string	Reserved to OpenSkyGems
schedreject	string	Last Scheduling Reject reason, empty equal to not elaborated or constraints ok
TKey	string	Search Key for Dynamic Target (Voyager CTNAME Field for Asteroid as for dynamic target sample CSV received). Match is only with this field !
TName	String	Designation Name of the Dynamic Object. Just an info, will not be used for search but is useful for who use the RoboTarget Manager to have in TName field of the Target a better name than the TKey string
TType	Integer	0=DSO/Default , 1=Comet , 2=Asteroid, 3=Planet, 4=DynaSearch . Using value of 1,2,3,4 the target is declared by Voyager Dynamic and Voyager will use the TKey for search the object in RoboOrbits and calculate the new cords RA/DEC when requested. MUST VALORIZED ever !
IsDynamicPointingOverride	Boolean	True if you want override the dynamic pointing mode of the base

		sequence for the target. Otherwise Voyager will use the Dynamic Pointing mode configured in the base sequence
DynamicPointingOverride	Integer	If IsDynamicPointingOverride is true you can define when Voyager will calculate with RoboOrbits then RA/DEC of the target. This Dynamic Pointing mode to use to override the base sequence. Values are 0=Begin of Sequence, 1= Each Goto in the Sequence, 2= Each X Seconds
DynEachX_Seconds	Integer	If you have defined the Each X Seconds DynamicPointingOverride you can override the number of seconds for the interval
DynEachX_Realign	Boolean	If you have defined the Each X Seconds DynamicPointingOverride you can override if Voyager will realign the target as soon as possible when the X seconds is passed by
DynEachX_NoPlateSolve	boolean	If you have defined the Each X Seconds DynamicPointingOverride and select to Realign you can define if use the plate solving for the pointing during the realign or just do a realign goto
isoffsetrf	boolean	True if the override of the offset steps adding to the final focus is enabled
offsetrf	integer	Number of steps to add to the final focus

Mask Char	Description
A	Position Angle
B	Min Altitude
C	Min SQM
D	HA Start

E	HA End
F	Date Start
G	Date End
H	Time Start
J	Time End
K	Moon Down
L	Moon Phase Min
M	Moon Phase Max
N	Moon Distance
O	HFD Mean Max
P	Max Shot Time For Day
Q	Airmass Min
R	Airmass Max
S	Max Time for sequence
T	OneShot Target

Mask 2 String	Description
L01	Moon Phase Min And Moon Up
M01	Moon Phase Max Or Moon Down
N01	Moon Distance or Moon Down
S01	Moon Lorentzian Avoidance or Moon Down

Lorentzian Profile	Description
0	Broad Band
1	Narrow Band
2	Free

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetTarget", "params": {"RefGuidSet":"f5421b76-40e4-4f06-82fb-b10f76b49295","UID":"1bc361c9-4fa4-4e47-a7a3-11172683b866","MAC":"ObQJIvIEdKfTPll6zc9mhfU8UpQ="}, "id": 15}

🡐{"jsonrpc": "2.0", "result": 0, "id": 15}

🡐

{"Event":"RemoteActionResult","Timestamp":1652224211.16795,"Host":"ORIONE","Inst":1,"UID":"1bc361c9-4fa4-4e47-a7a3-11172683b866","ActionResultInt":4,"Motivo":"","ParamRet":{"list":[{ "guid":"15b3a6cd-7a40-4cc7-8be9-d7d37d63a05d", "targetname":"IC1805", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"d80a07f9-c56e-4a42-bbc9-58a84c9a3438", "raj2000":2.54733333333333, "decj2000":61.471, "pa":0, "datecreation":1641751318, "status":0, "statusop":2, "priority":1, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"TestRotatoreMeridiano.s2q", "cid":"", "cmask":"BLN", "caltmin":40, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":0, "cmoondistance":35, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{

"guid":"ec4db992-402b-4502-9e4f-5138556a5fa6", "targetname":"IC447", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"90ae5721-a248-4159-ad74-56e13cf26141", "raj2000":6.52482527777778, "decj2000":9.97713333333333, "pa":180, "datecreation":1642032026, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"TestUnguidedNoPlateSolve.s2q", "cid":"", "cmask":"ADEMP", "caltmin":0, "csqmmin":0, "chastart":-2.5, "chaend":2.5, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":30, "cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":120, "cairmassmin":0, "cairmassmax":0, "cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"f9fbfd82-503a-4271-b0d4-56622692d84f", "targetname":"M101", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":14.0534952777778, "decj2000":54.34875, "pa":0, "datecreation":1640714968, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"12", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"DEM", "caltmin":0, "csqmmin":0, "chastart":-3, "chaend":3, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35, "cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"2166ed70-23d9-4668-bbbc-7a600a52ed6a", "targetname":"M101", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":14.0534952777778, "decj2000":54.34875, "pa":0, "datecreation":1640715122, "status":0, "statusop":2, "priority":1, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"BLMN", "caltmin":45, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":100, "cmoondistance":30, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"d6d991e9-98fb-4208-9faf-519c04ea33c6", "targetname":"M106", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":12.3161111111111, "decj2000":47.3037194444444, "pa":0, "datecreation":1642031395, "status":0, "statusop":2, "priority":1, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"BLMN", "caltmin":45, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":100, "cmoondistance":30, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"95cbe5f9-4a56-4af8-b3e6-26b0bf2bb7c8", "targetname":"M106", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":12.3161111111111, "decj2000":47.3037194444444, "pa":0, "datecreation":1642031553, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"DEK", "caltmin":0, "csqmmin":0, "chastart":-3, "chaend":3, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0, "cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"703d992c-293b-4fdb-a08b-f4e27176c5d6", "targetname":"M106Coppia", "refguidset":"f5421b76-

40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":12.2848672222222, "decj2000":47.2359833333333, "pa":180, "datecreation":1643541256,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"20", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"ADEM", "caltmin":0, "csqmmin":0, "chastart":-3, "chaend":3, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"ae4d20df-0b73-495e-832a-d1fe3a9d4af9", "targetname":"M106Coppia RGB",
"refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-
bbc4-655c401b05a4", "raj2000":12.2848672222222, "decj2000":47.2359833333333, "pa":180,
"datecreation":1647813658, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true",
"repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y",
"sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"ABLMN", "caltmin":45,
"csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0,
"cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":100, "cmoondistance":30,
"chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false,
"cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"9d35c5c3-bebf-4616-b2c7-
d79829ede800", "targetname":"M109", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295",
"refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":11.327155,
"decj2000":13.0446861111111, "pa":180, "datecreation":1646342995, "status":0, "statusop":2,
"priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"ADEM", "caltmin":0, "csqmmin":0, "chastart":-3.5, "chaend":3.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"14218175-92ae-4c48-a427-8671bc06bf1a", "targetname":"M34", "refguidset":"f5421b76-40e4-
4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":2.70206666666667, "decj2000":42.722, "pa":0, "datecreation":1642297968, "status":0,
"statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEMN", "caltmin":0, "csqmmin":0, "chastart":-2, "chaend":3, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":60,
"cmoondistance":45, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"0909dcaf-c032-4b1f-a099-df2d32c56d44", "targetname":"M42", "refguidset":"f5421b76-40e4-
4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":5.58813888888889, "decj2000":-5.39111111111111, "pa":0, "datecreation":1639855030,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEK", "caltmin":0, "csqmmin":0, "chastart":-2, "chaend":2, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"6ba9eb93-3ee3-4cf7-a070-f78964313b52", "targetname":"M78", "refguidset":"f5421b76-40e4-

4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":5.77916666666667, "decj2000":0.0791666666666667, "pa":0, "datecreation":1640715921,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEKP", "caltmin":0, "csqmmin":0, "chastart":-1.5, "chaend":1.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":120, "cairmassmin":0, "cairmassmax":0,
"cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"29076df0-458d-4c64-8588-6fbd059333a7", "targetname":"M78 RGB", "refguidset":"f5421b76-
40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":5.77916666666667, "decj2000":0.0791666666666667, "pa":0, "datecreation":1643893446,
"status":0, "statusop":2, "priority":3, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEMP", "caltmin":0, "csqmmin":0, "chastart":-2.5, "chaend":2.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":120, "cairmassmin":0, "cairmassmax":0,
"cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"0a4cad06-7ce7-42a9-822f-5fcdecc3e7bf", "targetname":"M81", "refguidset":"f5421b76-40e4-4f06-
82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":9.92583333333333, "decj2000":69.0652777777778, "pa":0, "datecreation":1640714886,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEK", "caltmin":0, "csqmmin":0, "chastart":-2.5, "chaend":2.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"65e03899-5ba8-4319-865c-776a949612ad", "targetname":"M81 RGB", "refguidset":"f5421b76-
40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":9.92583333333333, "decj2000":69.0652777777778, "pa":0, "datecreation":1643893746,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEM", "caltmin":0, "csqmmin":0, "chastart":-4.5, "chaend":4.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"0ab85aaf-b2b9-4ecf-9653-52a1a9920d2f", "targetname":"M82", "refguidset":"f5421b76-40e4-
4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",
"raj2000":9.93111111111111, "decj2000":69.6794444444444, "pa":0, "datecreation":1643501856,
"status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished",
"profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"",
"cmask":"DEM", "caltmin":0, "csqmmin":0, "chastart":-2.5, "chaend":2.5, "cdatestart":0, "cdateend":0,
"ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":0, "cmoonphasemax":35,
"cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0,
"cmask2":"M01", "cL01":false, "cM01":true, "cN01":false, "auxsessioncontainer":{"progress":100} },{
"guid":"155789cc-3c2d-4e78-85d8-b28617d808d2", "targetname":"NGC3628", "refguidset":"f5421b76-
40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4",

"raj2000":11.3380555555556, "decj2000":13.5894444444444, "pa":0, "datecreation":1642031608, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"DEK", "caltmin":0, "csqmmin":0, "chastart":-3, "chaend":3, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0, "cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"dc642b54-b0d4-419e-a147-64acfa2d0257", "targetname":"Rosetta Narrow", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":6.53253027777778, "decj2000":4.97793611111111, "pa":0, "datecreation":1639854793, "status":0, "statusop":2, "priority":1, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"BLMN", "caltmin":30, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":100, "cmoondistance":30, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"2e4f1737-932e-414b-a223-173797e4b0f2", "targetname":"SH2-174", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":23.7948341666667, "decj2000":80.9774972222222, "pa":180, "datecreation":1643505847, "status":0, "statusop":2, "priority":1, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"ABLMN", "caltmin":30, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":false, "cmoonphasemin":30, "cmoonphasemax":100, "cmoondistance":30, "chfdmeanlimit":0, "cmaxtimeforday":0, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} },{ "guid":"c5368274-86f6-48f2-a6fc-5f4837e06e1c", "targetname":"TapDoleNebula", "refguidset":"f5421b76-40e4-4f06-82fb-b10f76b49295", "refguidbasesequence":"24e4fcd5-a14d-41b3-bbc4-655c401b05a4", "raj2000":5.37521555555556, "decj2000":33.3692, "pa":0, "datecreation":1640813019, "status":0, "statusop":2, "priority":2, "note":"", "isrepeat":"true", "repeat":"6", "setname":"Finished", "profilename":"TestFlatNoMount.v2y", "sequencename":"SequenzaBase_TestFlatNoMount.s2q", "cid":"", "cmask":"BKP", "caltmin":50, "csqmmin":0, "chastart":0, "chaend":0, "cdatestart":0, "cdateend":0, "ctimestart":0, "ctimeend":0, "cmoondown":true, "cmoonphasemin":0, "cmoonphasemax":0, "cmoondistance":0, "chfdmeanlimit":0, "cmaxtimeforday":180, "cairmassmin":0, "cairmassmax":0, "cmask2":"", "cL01":false, "cM01":false, "cN01":false, "auxsessioncontainer":{"progress":100} }]}}

e) RemoteRoboTargetGetRunList

Method	`RemoteRoboTargetGetRunList`		
Description	Return list of RoboTarget Run for the Profile defined in Profile Parameter from Voyager ordered by datetime		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key

<table>
<tr><td></td><td>
<table>
<tr><td></td><td></td><td>string generated</td></tr>
<tr><td>ProfileName</td><td>String</td><td>Profile name used for search about Runs. If empty will be answered the Runs for all profile configured in Voyager</td></tr>
<tr><td>Days</td><td>numeric</td><td>Numer of days backward to today to search. If 0 days are used all the runs will be listed</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td>Integer(0)</td></tr>
<tr><td>License Required</td><td>Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>
<table>
<tr><td>List</td><td>Array</td><td>Array of Run Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>profilename</td><td>String</td><td>Named of profile name which Base Sequence belong</td></tr>
<tr><td>datetimestart</td><td>datetime</td><td>Run Start</td></tr>
<tr><td>datetimeend</td><td>datetime</td><td>Run End</td></tr>
<tr><td>isrunning</td><td>Boolean</td><td>True if the Run is running</td></tr>
<tr><td>seqcount</td><td>Integer</td><td>Count of Sequences done in Run</td></tr>
<tr><td>errcount</td><td>Integer</td><td>Count of Errors thrown during the Run</td></tr>
<tr><td>note</td><td>string</td><td>Text of RoboTarget settings used for the Run</td></tr>
</table>
</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetRunList”, “params”: {“ProfileName”:”TestFlatNoMount.v2y”,”Days”:30,”UID”:”86991d3d-5e53-4611-8d18-d9de253e4c52”,”MAC”:”fiigmm8z80M2Xid7oG04EHRKjVA=”}, “id”: 14}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 14}

➔
{“Event”:”RemoteActionResult”,”Timestamp”:1652619644.23514,”Host”:”ORIONE”,”Inst”:1,”UID”:”86991d3d-5e53-4611-8d18-d9de253e4c52”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”72af54ea-bd48-4661-b553-eecf58bb1600”, “profilename”:”TestFlatNoMount”, “datetimestart”:1652013635, “datetimeend”:1652013635, “isrunning”:false, “seqcount”:0, “errcount”:1,

“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=ASTRONOMICAL;OffsetMinutesStart=-180;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=False;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”1fb7d575-90°2-4391-a81f-60fe55b11169”, “profilename”:”TestFlatNoMount”, “datetimestart”:1652013616, “datetimeend”:1652013616, “isrunning”:false, “seqcount”:0, “errcount”:1, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=ASTRONOMICAL;OffsetMinutesStart=-180;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=False;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”f04b7b84-ad13-4209-b9d5-34c89fb5b97a”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651929309, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:5, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”bf986992-a362-4174-85d7-13036da0ddfe”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651928960, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:3, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”70ea3189-bbce-428f-a7ec-bdc6af72983e”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651928903, “datetimeend”:1651928955, “isrunning”:false, “seqcount”:0, “errcount”:4, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”a87f0a6d-9895-409b-9c9a-b1ba0e8c357b”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651928832, “datetimeend”:1651928884, “isrunning”:false, “seqcount”:0, “errcount”:1, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”8b7d6879-9335-4496-9fc4-f28c47bb42f8”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651925029, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:4, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”56d3a2d2-2°3f-4ebb-9428-494f324cd640”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651925015, “datetimeend”:1651925022, “isrunning”:false, “seqcount”:0, “errcount”:1, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”b6b2d872-d2fa-4bda-8e6f-d39679d8043b”, “profilename”:”TestFlatNoMount”, “datetimestart”:1651925011, “datetimeend”:1651925011, “isrunning”:false, “seqcount”:0, “errcount”:1, “note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{ “guid”:”8ccc9a18-72f9-4°11-9b77-b11d422d8cae”, “profilename”:”TestFlatNoMount”,

“datetimestart”:1651924840, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:3,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”257c52d1-2c1d-476b-a788-92bfc8b39e60”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924780, “datetimeend”:1651924837, “isrunning”:false, “seqcount”:0, “errcount”:3,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”2b074240-3224-4f06-9445-bc7ff02c0acf”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924768, “datetimeend”:1651924776, “isrunning”:false, “seqcount”:0, “errcount”:1,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”51ed8624-ede9-4afe-9262-91cccdf3e783”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924616, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:1,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”bc874d20-a26f-4f44-a5c3-a7de6be22302”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924543, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:3,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”2afe19a3-120e-49°1-b04a-d59478fa44f2”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924509, “datetimeend”:1651924532, “isrunning”:false, “seqcount”:0, “errcount”:2,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”a2099dfe-ccdf-40d5-a604-e23ef1288f5d”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651924486, “datetimeend”:1651924499, “isrunning”:false, “seqcount”:0, “errcount”:1,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”0e4dd276-6bb0-49af-91df-854°727a2d8e”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651920763, “datetimeend”:1651920812, “isrunning”:false, “seqcount”:0, “errcount”:2,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”8333183f-d8ff-4845-8505-f464815a2d7b”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651920751, “datetimeend”:1651920760, “isrunning”:false, “seqcount”:0, “errcount”:0,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;Offs
etMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMax
ShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”931f4d83-e7f6-41ce-a271-d20409a6dc00”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651920408, “datetimeend”:0, “isrunning”:false, “seqcount”:0, “errcount”:2,

“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”bcd9561e-a371-4530-9e47-dfb599a60efa”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651920293, “datetimeend”:1651920309, “isrunning”:false, “seqcount”:0, “errcount”:2,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”8dd6004c-5ba7-418d-99b8-17efbcb216c4”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651920031, “datetimeend”:1651920081, “isrunning”:false, “seqcount”:0, “errcount”:2,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”8ea4d45d-1729-4°17-9f7a-7d2369d3f7c6”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651919947, “datetimeend”:1651919958, “isrunning”:false, “seqcount”:0, “errcount”:0,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”372574c8-4bca-4358-8794-6d53e0fa8e84”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651919784, “datetimeend”:1651919787, “isrunning”:false, “seqcount”:0, “errcount”:1,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”e7742a25-195f-42d9-9bd9-dccac88e4dba”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651919001, “datetimeend”:1651919245, “isrunning”:false, “seqcount”:0, “errcount”:0,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” },{
“guid”:”47e83a6f-2d14-4197-8094-acb6b2c4bb01”, “profilename”:”TestFlatNoMount”,
“datetimestart”:1651918266, “datetimeend”:1651918294, “isrunning”:false, “seqcount”:0, “errcount”:1,
“note”:”;Scheduler=DefaultScheduler;SchedulerVer=1.0.0;RobotargetActionVer=1.0.0;NightType=CIVIL;OffsetMinutesStart=0;OffsetMinutesEnd=0;SequenceMinDuration=15;MoonDownAltitude=0;NoTargetUseMaxShotSeq=True;SoftRetry=5;SequenceMaxRetry=5;NoSequenceInMeridianNoGotoZone=True” }]}}

f) RemoteRoboTargetGetShot

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetShot</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Shot associated to the Target</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidTarget</td><td>String</td><td>UID of Target used for search</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>Array</td><td>Array of Shot Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>label</td><td>String</td><td>Label to use in addition to filename</td></tr>
<tr><td>refguidtarget</td><td>datetime</td><td>UID of Target</td></tr>
<tr><td>filterindex</td><td>datetime</td><td>See RemoteCameraShot command</td></tr>
<tr><td>num</td><td>integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>bin</td><td>Integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>readoutmode</td><td>Integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>type</td><td>integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>speed</td><td>integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>gain</td><td>integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>offset</td><td>integer</td><td>See RemoteCameraShot command</td></tr>
<tr><td>exposure</td><td>numeric</td><td>See RemoteCameraShot command</td></tr>
<tr><td>order</td><td>integer</td><td>Execution order</td></tr>
<tr><td>done</td><td>boolean</td><td>Not used</td></tr>
<tr><td>enabled</td><td>boolean</td><td>True if enabled</td></tr>
<tr><td>auxtotshot</td><td>integer</td><td>Number of total shot to do included the repeat if configured</td></tr>
<tr><td>auxshotdone</td><td>integer</td><td>Number of Shot</td></tr>
</table></td></tr>
</table>

					done
			auxshotdonedeleted	integer	Number of shot done and logically removed

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetShot”, “params”: {“RefGuidTarget”:”632200ce-2145-4295-9236-0c459b3ac196”,”UID”:”45°77a8e-9e0f-4da5-b185-d48d1f5a9847”,”MAC”:”boIBnnM9Cq8aHCrJfRM5G/5FKY8=”}, “id”: 15}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 15}

➔

{“Event”:”RemoteActionResult”,”Timestamp”:1652620278.2191,”Host”:”ORIONE”,”Inst”:1,”UID”:”45°77a8e-9e0f-4da5-b185-d48d1f5a9847”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”be07deef-9216-4b88-a25d-4bbe6748794e”, “label”:””, “refguidtarget”:”632200ce-2145-4295-9236-0c459b3ac196”, “filterindex”:4, “num”:12, “bin”:1, “readoutmode”:0, “type”:0, “speed”:0, “gain”:1600, “offset”:27, “exposure”:600, “order”:1, “done”:false, “enabled”:true, “auxtotshot”:72, “auxshotdone”:70, “auxshotdonedeleted”:0},{ “guid”:”14dca058-f03e-46e1-8°00-1d9f55c061c4”, “label”:””, “refguidtarget”:”632200ce-2145-4295-9236-0c459b3ac196”, “filterindex”:5, “num”:12, “bin”:1, “readoutmode”:0, “type”:0, “speed”:0, “gain”:1600, “offset”:27, “exposure”:600, “order”:2, “done”:false, “enabled”:true, “auxtotshot”:72, “auxshotdone”:60, “auxshotdonedeleted”:0}]}}

g) RemoteRoboTargetGetSequenceListByProfile

Method	RemoteRoboTargetGetSequenceListByProfile			
Description	Return list of Sequence available for a profile in default Voyager Sequence configuration folder			
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated	
	ProfileName	String	Profile name to use for the search. Cannot be empty.	
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.	
Result	Integer(0)			
License Required	*Advanced, Full*			
Remote Action Result Parameters in	List	Array	Array of Sequence Objects	
			name	String	Sequence name

ParamRet Object			filename	string	Sequence file with path
			profilename	string	Profile name associated to the sequence

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetSequenceListByProfile”, “params”: {“ProfileName”:”TestFlatNoMount.v2y”,”UID”:”98129170-e267-4f8b-9°21-4e773b2889de”,”MAC”:”su2SH/Bq9aExUKd0BWJkKzfBFy0=”}, “id”: 22}

🡐{“jsonrpc”: “2.0”, “result”: 0, “id”: 22}

🡐
{“Event”:”RemoteActionResult”,”Timestamp”:1652626905.66952,”Host”:”ORIONE”,”Inst”:1,”UID”:”98129170-e267-4f8b-9°21-4e773b2889de”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “name”:”SequenzaBase_TestFlatNoMount.s2q”, “filename”:”C:\\Users\\pegas\\OneDrive\\Documenti\\Voyager\\ConfigSequence\\SequenzaBase_TestFlatNoMount.s2q”, “profilename”:”TestFlatNoMount.v2y” },{ “name”:”TestRotatoreMeridiano.s2q”, “filename”:”C:\\Users\\pegas\\OneDrive\\Documenti\\Voyager\\ConfigSequence\\TestRotatoreMeridiano.s2q”, “profilename”:”TestFlatNoMount.v2y” },{ “name”:”TestUnguidedNoPlateSolve.s2q”, “filename”:”C:\\Users\\pegas\\OneDrive\\Documenti\\Voyager\\ConfigSequence\\TestUnguidedNoPlateSolve.s2q”, “profilename”:”TestFlatNoMount.v2y” }]}}

h) RemoteRoboTargetGetSessionListByRun

Method	RemoteRoboTargetGetSessionListByRun			
Description	Return list of RoboTarget Session done during the Run			
Params	UID		String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidRun		String	UID of Run.
	MAC		String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)			
License Required	*Advanced, Full*			
Remote Action Result Parameters in	List	Array	Array of Session Objects	

ParamRet Object			guid	String	UID of Session
			datetimestart	datetime	Datetime of the Session start
			datetimeend	datetime	Datetime of the Session end
			repfilepdf	String	PDF report file if present
			refguidrun	string	UID of Run which session belong
			refguidtarget	string	UID of Target shot during Session
			result	integer	Session result, see table below
			status	integer	Session status 0=Idle 1=Running
			targetname	string	Name of the Target done during the session
			shotnumber	integer	Shot done for this session
			shotnumberdeleted		Shot done and delete for this session

Session Result	Description
0 = UNDEF	Undefined
1 = OK	Session finished without error
2 = ABORTED	Session aborted
3 = FINISHED_ERROR	Session finished with error
4 = TIMEOUT	Session finished for timeout

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetSessionListByRun”, “params”: {“RefGuidRun”:”98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1”,”UID”:”f1c9a590-cf9b-4892-9049-84e648241d47”,”MAC”:”M7luxHQUkzJTktO7r70Ef1OiZzI=”}, “id”: 34}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 34}

➔

{“Event”:”RemoteActionResult”,”Timestamp”:1652628087.20301,”Host”:”RC16”,”Inst”:1,”UID”:”f1c9a590-cf9b-4892-9049-84e648241d47”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”ee1441e6-8db7-49d7-a715-2487f988b725”, “datetimestart”:1651095781, “datetimeend”:1651106015, “repfilepdf”:””, “refguidrun”:”98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1”, “refguidtarget”:”6b26f367-b2b9-

4987-b2d8-d3a975f072a6", "result":3, "status":0, "targetname":"M109 RGB", "shotnumber":24, "shotnumberdeleted":0 },{ "guid":"fbf3b7c3-e0a4-418c-9e41-466b50ef9b40", "datetimestart":1651106083, "datetimeend":1651106493, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"ed0a329b-dcd1-449e-b861-7afc32b55cd6", "result":3, "status":0, "targetname":"Doppietto Leone RGB", "shotnumber":0, "shotnumberdeleted":0 },{ "guid":"a87fb0a4-79e6-4ba5-b74e-eed552dd9fee", "datetimestart":1651106560, "datetimeend":1651108923, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"5f5ead07-1ad6-4172-b4e0-2cc8f7b42ef5", "result":1, "status":0, "targetname":"2Galassie", "shotnumber":1, "shotnumberdeleted":0 },{ "guid":"5f684a79-f1e5-49c5-b8ee-783398c70b4d", "datetimestart":1651108931, "datetimeend":1651109498, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"6b26f367-b2b9-4987-b2d8-d3a975f072a6", "result":3, "status":0, "targetname":"M109 RGB", "shotnumber":0, "shotnumberdeleted":0 },{ "guid":"a0d35a84-90b1-42e8-9085-43510ebcff74", "datetimestart":1651109566, "datetimeend":1651114383, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"bc7387b5-2f4e-4c71-8055-e66e118383ec", "result":1, "status":0, "targetname":"NGC4725", "shotnumber":12, "shotnumberdeleted":0 },{ "guid":"f049c971-3b06-4b99-ba57-75965fee6c92", "datetimestart":1651114391, "datetimeend":1651116723, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"0c646551-c271-49e8-b7ab-7212febb908d", "result":1, "status":0, "targetname":"M51", "shotnumber":5, "shotnumberdeleted":0 },{ "guid":"8c723251-3°84-4862-8cca-2d78be382ff8", "datetimestart":1651116731, "datetimeend":1651120083, "repfilepdf":"", "refguidrun":"98f2d1bb-3fa4-45°2-b369-5fbbc6f8baf1", "refguidtarget":"571204ae-1d0c-4318-a685-b0583b6df0fd", "result":1, "status":0, "targetname":"NGC5907", "shotnumber":8, "shotnumberdeleted":0 }]}}

i) RemoteRoboTargetGetShotDoneBySessionList

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetShotDoneBySessionList</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Shot done during the Session</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidSession</td><td>String</td><td>UID of Run.</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table></td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in</td><td colspan="3"><table>
<tr><td>List</td><td>object</td><td>Array of Shot Done Objects divided in 2 categories (done and deleted) with the same structure</td></tr>
</table></td></tr>
</table>

ParamRet Object			guid	String	UID of Shot Done
			datetimeshot	datetime	Datetime of the shot
			datetimeshotutc	Datetime	Date time UTC when shot was done
			filename	string	Filename of the shot
			hfd	numeric	Average HFD in pixel
			max	numeric	Max ADU value of a pixel in the image
			mean	numeric	Average ADU value of a pixel in the image
			min	numeric	Min ADU value of a pixel in the image
			path	string	Shot file path
			refguidsession	string	UID of Session wich shot belong
			refguidshot	string	UID of Shot which this shot inherits
			starindex	numeric	Star index of the shot image
			bin	integer	Binning used for image
			filterindex	integer	Filter index of the image
			exposure	numeric	Exposure in seconds
			rating	integer	Rating value in integer

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetShotDoneBySessionList”, “params”: {“RefGuidSession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”MAC”:”T5/w3jjW1LsnloyymlWeUi4Z0OI=”}, “id”: 38}

←{“jsonrpc”: “2.0”, “result”: 0, “id”: 38}

←

{“Event”:”RemoteActionResult”,”Timestamp”:1652628775.2153,”Host”:”RC16”,”Inst”:1,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{ “list”:{“done”:[{ “guid”:”a062fe95-fc96-4e49-ba7e-2bc5dfd9d105”, “datetimeshot”:1651106944, “filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_001_20220428_004904_471_GA_1087_OF_60_E.FIT”, “hfd”:4.75, “max”:65535, “mean”:5586, “min”:5032, “path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-27\\L”,

“refguidsession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.33, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 }],”deleted”:[]} }}

j) RemoteRoboTargetGetShotDoneBySetList

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetShotDoneBySetList</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Shot done for a Set</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidSet</td><td>String</td><td>UID of Set.</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>object</td><td>Array of Shot Done Objects divided in 2 categories (done and deleted) with the same structure
<table>
<tr><td>guid</td><td>String</td><td>UID of ShotDone</td></tr>
<tr><td>datetimeshot</td><td>datetime</td><td>Datetime of the shot</td></tr>
<tr><td>datetimeshotutc</td><td>Datetime</td><td>Date time UTC when shot was done</td></tr>
<tr><td>filename</td><td>string</td><td>Filename of the shot</td></tr>
<tr><td>hfd</td><td>numeric</td><td>Average HFD in pixel</td></tr>
<tr><td>max</td><td>numeric</td><td>Max ADU value of a pixel in the image</td></tr>
<tr><td>mean</td><td>numeric</td><td>Average ADU value of a pixel in the image</td></tr>
<tr><td>min</td><td>numeric</td><td>Min ADU value of a pixel in the image</td></tr>
<tr><td>path</td><td>string</td><td>Shot file path</td></tr>
<tr><td>refguidsession</td><td>string</td><td>UID of Session wich shot belong</td></tr>
<tr><td>refguidshot</td><td>string</td><td>UID of Shot which this shot inherits</td></tr>
<tr><td>starindex</td><td>numeric</td><td>Star index of the</td></tr>
</table>
</td></tr>
</table>

					shot image
			bin	integer	Binning used for image
			filterindex	integer	Filter index of the image
			exposure	numeric	Exposure in seconds
			rating	integer	Rating value in integer

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetShotDoneBySetList”, “params”: {“RefGuidSet”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”MAC”:”T5/w3jjW1LsnloyymlWeUi4Z0OI=”}, “id”: 38}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 38}

➔
{“Event”:”RemoteActionResult”,”Timestamp”:1652628775.2153,”Host”:”RC16”,”Inst”:1,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{ “list”:{“done”:[{ “guid”:”a062fe95-fc96-4e49-ba7e-2bc5dfd9d105”, “datetimeshot”:1651106944, “filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_001_20220428_004904_471_GA_1087_OF_60_E.FIT”, “hfd”:4.75, “max”:65535, “mean”:5586, “min”:5032, “path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-27\\L”, “refguidsession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.33, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 }],”deleted”:[]} }}

k) RemoteRoboTargetGetShotDoneSinceList

Method	RemoteRoboTargetGetShotDoneSinceList			
Description	Return list of RoboTarget Shot done after a datetime (all, for Set or for Target)			
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated	
	Since	datetime	Epoch format of datetime where the search start	
	SinceUTC	Datetime	Epoch format of datetime UTC where the search start	
	RefGuidSet	String	UID of Set. Empty for all Set also Shot will reported only for the Set selected. Mutual exclusive with RefGuidTarget (use RegGuidTarget or RefGuidSet).	
	RefGuidTarget	String	UID of Target. Empty for all Target also Shot will reported only for the Target selected. Mutual exclusive with RefGuidSet (use	

<table>
<tr><td></td><td></td><td></td><td>RegGuidTarget or RefGuidSet).</td></tr>
<tr><td></td><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>object</td><td>Array of Shot Done Objects divided in 2 categories (done and deleted) with the same structure
<table>
<tr><td>guid</td><td>String</td><td>UID of Shot Done</td></tr>
<tr><td>datetimeshot</td><td>datetime</td><td>Datetime of the shot</td></tr>
<tr><td>datetimeshotutc</td><td>Datetime</td><td>Date time UTC when shot was done</td></tr>
<tr><td>filename</td><td>string</td><td>Filename of the shot</td></tr>
<tr><td>hfd</td><td>numeric</td><td>Average HFD in pixel</td></tr>
<tr><td>max</td><td>numeric</td><td>Max ADU value of a pixel in the image</td></tr>
<tr><td>mean</td><td>numeric</td><td>Average ADU value of a pixel in the image</td></tr>
<tr><td>min</td><td>numeric</td><td>Min ADU value of a pixel in the image</td></tr>
<tr><td>path</td><td>string</td><td>Shot file path</td></tr>
<tr><td>refguidset</td><td>string</td><td>UID of Set which Shot belong</td></tr>
<tr><td>refguidtarget</td><td>string</td><td>UID of Target which Shot belong</td></tr>
<tr><td>refguidsession</td><td>string</td><td>UID of Session wich shot belong</td></tr>
<tr><td>refguidshot</td><td>string</td><td>UID of Shot which this shot inherits</td></tr>
<tr><td>starindex</td><td>numeric</td><td>Star index of the shot image</td></tr>
<tr><td>bin</td><td>integer</td><td>Binning used for image</td></tr>
<tr><td>filterindex</td><td>integer</td><td>Filter index of the image</td></tr>
<tr><td>exposure</td><td>numeric</td><td>Exposure in seconds</td></tr>
</table></td></tr>
</table>

			rating	integer	Rating value in integer
			raj2000	string	RA J2000 string format of target
			decj2000	string	DEC J2000 string format of target

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetShotDoneSinceList”, “params”: {“RefGuidSet”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”Since”:1652331631,”RefGuidTarget”:””,”RefGuidSet”:””,”MAC”:”T5/w3jjW1LsnloyymlWeUi4Z0OI=”}, “id”: 38}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 38}

➔
{“Event”:”RemoteActionResult”,”Timestamp”:1652628775.2153,”Host”:”RC16”,”Inst”:1,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{ “list”:{“done”:[{ “guid”:”a062fe95-fc96-4e49-ba7e-2bc5dfd9d105”, “datetimeshot”:1651106944, “filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_001_20220428_004904_471_GA_1087_OF_60_E.FIT”, “hfd”:4.75, “max”:65535, “mean”:5586, “min”:5032, “path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-27\\L”, “refguidtarget”:”a87fb0a4-79e6-4ba5-b74e-eed5534d9fee”, “refguidset”:”a835644-79e6-4ba5-b74e-eed552dd9fee”, “refguidsession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.33, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 }],”deleted”:[]} }}

l) RemoteRoboTargetGetShotDoneBySlotList

Method	`RemoteRoboTargetGetShotDoneBySlotList`
Description	Return list of RoboTarget Shot done for a Slot .
Params	UID — String — Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated RefGuidShot — String — UID of Shot. MAC — String — Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)

<table>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>object</td><td>Array of Shot Done Objects divided in 2 categories (done and deleted) with the same structure

<table>
<tr><td>guid</td><td>String</td><td>UID of Shot Done</td></tr>
<tr><td>datetimeshot</td><td>datetime</td><td>Datetime of the shot</td></tr>
<tr><td>datetimeshotutc</td><td>Datetime</td><td>Date time UTC when shot was done</td></tr>
<tr><td>filename</td><td>string</td><td>Filename of the shot</td></tr>
<tr><td>hfd</td><td>numeric</td><td>Average HFD in pixel</td></tr>
<tr><td>max</td><td>numeric</td><td>Max ADU value of a pixel in the image</td></tr>
<tr><td>mean</td><td>numeric</td><td>Average ADU value of a pixel in the image</td></tr>
<tr><td>min</td><td>numeric</td><td>Min ADU value of a pixel in the image</td></tr>
<tr><td>path</td><td>string</td><td>Shot file path</td></tr>
<tr><td>refguidsession</td><td>string</td><td>UID of Session wich shot belong</td></tr>
<tr><td>refguidshot</td><td>string</td><td>UID of Shot which this shot inherits</td></tr>
<tr><td>starindex</td><td>numeric</td><td>Star index of the shot image</td></tr>
<tr><td>bin</td><td>integer</td><td>Binning used for image</td></tr>
<tr><td>filterindex</td><td>integer</td><td>Filter index of the image</td></tr>
<tr><td>exposure</td><td>numeric</td><td>Exposure in seconds</td></tr>
<tr><td>rating</td><td>integer</td><td>Rating value in integer</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetShotDoneBySessionList”, “params”: {“RefGuidSession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”,”UID”:”cdbc181d-be39-486c-adf2-50bbe313b0fb”,”MAC”:”T5/w3jjW1LsnloyymlWeUi4Z0OI=”}, “id”: 38}

➔{“jsonrpc”: “2.0”, “result”: 0, “id”: 38}

➔

{“Event”:”RemoteActionResult”,”Timestamp”:1652629848.93571,”Host”:”RC16”,”Inst”:1,”UID”:”b3e2cf9f-a906-4708-8733-949ad88e156c”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{ “list”:{“done”:[{

“guid”:”a062fe95-fc96-4e49-ba7e-2bc5dfd9d105”, “datetimeshot”:1651106944,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_001_20220428_004904_471_GA_1087_OF_60_E.FIT”,
“hfd”:4.75, “max”:65535, “mean”:5586, “min”:5032,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-27\\L”,
“refguidsession”:”a87fb0a4-79e6-4ba5-b74e-eed552dd9fee”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.33, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”53fbd41b-4928-4531-a5d5-81b04e41c6d2”, “datetimeshot”:1651185973,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_002_20220428_224613_251_GA_1087_OF_60_E.FIT”,
“hfd”:5.48, “max”:65535, “mean”:5318, “min”:4816,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.56, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”ff3bbed1-91b0-40af-a665-263°3ca2e455”, “datetimeshot”:1651186463,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_003_20220428_225423_186_GA_1087_OF_60_E.FIT”,
“hfd”:5.58, “max”:65535, “mean”:5352, “min”:4736,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:7.03, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”db9a6729-d3c1-4952-90c4-71c6d2afc392”, “datetimeshot”:1651186782,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_004_20220428_225942_050_GA_1087_OF_60_E.FIT”,
“hfd”:5.36, “max”:65535, “mean”:5367, “min”:4808,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.65, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”b2cc3c10-0408-4dcb-b29e-ec88d675b9c0”, “datetimeshot”:1651187538,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_005_20220428_231218_710_GA_1087_OF_60_E.FIT”,
“hfd”:4.82, “max”:65535, “mean”:5404, “min”:4880,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.39, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”450afe5d-e457-4151-aa62-ec530b011037”, “datetimeshot”:1651187903,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_006_20220428_231823_712_GA_1087_OF_60_E.FIT”,
“hfd”:4.82, “max”:65535, “mean”:5417, “min”:4864,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.95, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”3132b14c-17°6-4242-a557-adc0fca562c6”, “datetimeshot”:1651188334,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_007_20220428_232534_072_GA_1087_OF_60_E.FIT”,
“hfd”:4.79, “max”:65535, “mean”:5433, “min”:4912,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.09, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”c12508f7-de33-46ed-956f-caceb1852840”, “datetimeshot”:1651188700,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_008_20220428_233140_121_GA_1087_OF_60_E.FIT”,
“hfd”:5.05, “max”:65535, “mean”:5451, “min”:4932,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,

“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.04, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”357d5ac1-b37e-48ed-b3bb-f4624e4482a9”, “datetimeshot”:1651189226,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_009_20220428_234026_446_GA_1087_OF_60_E.FIT”, “hfd”:4.8, “max”:65535, “mean”:5488, “min”:4924,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.86, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”e86caaa1-9303-42e3-b3fe-7ec1835f301a”, “datetimeshot”:1651189591,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_010_20220428_234631_073_GA_1087_OF_60_E.FIT”, “hfd”:5, “max”:65535, “mean”:5496, “min”:4964,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.24, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”4342dd1e-5b90-4e75-a518-a04441db113d”, “datetimeshot”:1651189893,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_001_20220428_235133_968_GA_1087_OF_60_E.FIT”, “hfd”:5.08, “max”:65535, “mean”:5495, “min”:4932,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.31, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”c26cc0b1-4870-4d61-9ce7-547128c649d9”, “datetimeshot”:1651190344,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_002_20220428_235904_405_GA_1087_OF_60_E.FIT”, “hfd”:4.85, “max”:65535, “mean”:5518, “min”:4912,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.24, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”05f2b7a5-b0ae-4493-879°-201fb85f8110”, “datetimeshot”:1651190674,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_003_20220429_000434_268_GA_1087_OF_60_E.FIT”, “hfd”:5.25, “max”:65535, “mean”:5543, “min”:4936,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.01, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”74eef5d3-7ad2-474c-9b82-208954eefb0e”, “datetimeshot”:1651190977,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_004_20220429_000937_242_GA_1087_OF_60_E.FIT”, “hfd”:5.21, “max”:65535, “mean”:5551, “min”:4972,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.76, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”694c61e7-a4b5-4630-9336-c67ca2dc2af6”, “datetimeshot”:1651191343,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_005_20220429_001543_213_GA_1087_OF_60_E.FIT”, “hfd”:5.04, “max”:65535, “mean”:5581, “min”:5000,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:6.08, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”07f5dbfb-d71d-4128-99bf-b31a71a59343”, “datetimeshot”:1651191646,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_006_20220429_002046_092_GA_1087_OF_60_E.FIT”,

“hfd”:5.15, “max”:65535, “mean”:5623, “min”:5020,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.75, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 },{ “guid”:”293e8e0e-f110-41d4-a603-ed3f633b3732”, “datetimeshot”:1651192161,
“filename”:”2Galassie_LIGHT_L_300s_BIN1_-12C_007_20220429_002921_542_GA_1087_OF_60_E.FIT”,
“hfd”:4.91, “max”:65535, “mean”:5649, “min”:5020,
“path”:”C:\\Users\\grissino\\Documents\\Voyager\\Sequence\\2Galassie\\2022-04-28\\L”,
“refguidsession”:”ed22fd79-aadb-4052-8bd9-a2fcd0e3d08c”, “refguidshot”:”48b1d49c-8dac-44e7-a72d-af3153e356c0”, “starindex”:5.66, “bin”:1, “filterindex”:0, “exposure”:300, “rating”:0 }],”deleted”:[]} }}

m) RemoteRoboTargetGetErrorListByRun

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetErrorListByRun</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Error done for a Run .</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidRun</td><td>String</td><td>UID of Run.</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3">
<table>
<tr><td>List</td><td>object</td><td>Array of Error Objects
<table>
<tr><td>guid</td><td>String</td><td>UID of Error</td></tr>
<tr><td>datetimestart</td><td>datetime</td><td>Datetime start error</td></tr>
<tr><td>errorcode</td><td>integer</td><td>Error code, see list below</td></tr>
<tr><td>note</td><td>string</td><td>Text about error</td></tr>
<tr><td>refguidrun</td><td>string</td><td>UID of Run</td></tr>
<tr><td>refguidsession</td><td>string</td><td>UID of Session if available</td></tr>
<tr><td>refguidtarget</td><td>string</td><td>UID of Target if available</td></tr>
<tr><td>targetname</td><td>string</td><td>Target name if available</td></tr>
<tr><td>seqerrorcode</td><td>integer</td><td>Sequence error code if available</td></tr>
</table>
</td></tr>
</table>
</td></tr>
</table>

			seqexectime	integer	Sequence duration time at error time
			seqshotaterror	integer	Sequence shot done at error time

Error Code	Description
0	No error
1	Unknow error
100	No shot configured for target during scheduling process
102	Error retrieving multiplier for shot during scheduling process
103	Cannot check Shot Progress during scheduling process
104	SQM control isn't available
105	Cannot calculate time to shot for finish a target
106	Cannon calculate time to shot
1000	Error during Scheduling Apply
1001	Error during Base Sequence loading
1002	No one shot configured in Database
1003	Cannot get shot data from DB
1004	Sequence cannot set the coordinates to sequence
1005	Sequence cannot start
1006	Exit for Error
1007	Wrong Voyager profile loaded, actual profile do not match the sequence profile
1008	Cannot get shot group multiplier
1009	Reached the max run count for a Sequence in a day
1010	Cannot start the sequence because the Target have an HA inside the no goto zone
1011	Emergency Suspend happens
1012	Emergency Exit happens
1013	The target is an orphan (or the set which the target belong are an orphan)
2000	The Sequence end with an error
2001	The Sequence end for timeout

```
NO_ERROR = 0
UNKNOW = 1

SCHED_NO_SHOT_CONFIGURED = 100
SCHED_SEQBASE_ACCESS = 101
SCHED_LOOP_MULTIPLIER = 102
SCHED_CHECK_SHOT_PROGRESS = 103
SCHED_SQM_UNAVAILABLE = 104
SCHED_TIME_TO_SHOT_FOR_FINISH = 105
SCHED_TIME_SHOT = 106

RT_SCHEDULER_APPLY = 1000
RT_CARICAMENTO_SEQUENZA_BASE = 1001
RT_SHOT_NO_ONE_IN_DB = 1002
RT_SHOT_CANNOT_GET_FROM_DB = 1003
RT_SEQUENCE_CANNOT_SET_COORDS = 1004
RT_SEQUENCE_CANNOT_START = 1005
RT_EXIT_ERROR = 1006
RT_WRONG_PROFILE = 1007
RT_CANNOT_GET_MULTIPLIER = 1008
RT_SEQUENCE_MAX_RETRY_FOR_RUN = 1009
```

```
RT_SEQUENCE_NO_GOTO_ZONE = 1010
RT_EMERGENCY_SUSPEND = 1011
RT_EMERGENCY_EXIT = 1012
RT_TARGET_ORPHANS = 1013

'2000-2999 Reserved to Sequence
SEQUENCE_END_ERROR = 2000
SEQUENCE_END_TIMEOUT = 2001
```

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetErrorListByRun", "params": {"RefGuidRun":"72af54ea-bd48-4661-b553-eecf58bb1600","UID":"1f78ce78-012b-427e-8ed8-7d811d9edb76","MAC":"FIIZSgTw5CXgZoldQlxGXRtvof4="}, "id": 11}

🡐{"jsonrpc": "2.0", "result": 0, "id": 11}

🡐

{"Event":"RemoteActionResult","Timestamp":1652626535.24723,"Host":"ORIONE","Inst":1,"UID":"1f78ce78-012b-427e-8ed8-7d811d9edb76","ActionResultInt":4,"Motivo":"","ParamRet":{"list":[{ "guid":"19aa7b52-46de-4d03-8349-2d8bb6b0246a", "datetimestart":1652013635, "errorcode":1006, "note":"Please Connect All Voyager Controls", "refguidrun":"72af54ea-bd48-4661-b553-eecf58bb1600", "refguidsession":"", "refguidtarget":"", "targetname":"", "seqerrcode":0, "seqexectime":0, "seqshotaterror":0 }]}}

n) RemoteRoboTargetGetAnnotationListByRun

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetAnnotationListByRun</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Error done for a Run .</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidRun</td><td>String</td><td>UID of Run.</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>object</td><td>Array of Annotation Objects
<table>
<tr><td>guid</td><td>String</td><td>UID of Annotation</td></tr>
<tr><td>datetimestart</td><td>datetime</td><td>Datetime start annotation</td></tr>
</table>
</td></tr>
</table>

			code	integer	Annotation code, see list below
			note	string	Text about error
			refguidrun	string	UID of Run
			refguidtarget	string	UID of Target if available
			targetname	string	Target name if available

Annotation Code	Description
0	No annotation
1	Unknow error
2	Suspend removed for a target
3	Shot Done removed
4	Shot Done removed fo all session
5	Rating Updated
6	Update Rating of all shot belong to session
7	Bulk remove of Shot
8	Bulk rating of shot
9	Restore Shot Done removed belong to the session
10	Restore shot done removed
11	Update rating of all target
12	Restore shot done for the target
13	Remove shot done for the target
14	Remove shot done for all shot belong to slot
15	Restore shot done for all shot belong to slot
16	Update rating of shot for all shot belong to slot

```
NO_ANNOTATION = 0

REMOVE_SUSPEND = 1
USER_ABORT_SEQUENCE = 2
REMOVE_SHOT_DONE = 3
REMOVE_SHOT_DONE_ALL_SESSION = 4
UPDATE_RATING = 5
UPDATE_RATING_ALL_SESSION = 6
UPDATEBULK_DELETE = 7
UPDATEBULK_RATING = 8
RESTORE_SHOT_DONE_ALL_SESSION = 9
RESTORE_SHOT_DONE = 10
UPDATE_RATING_ALL_TARGET = 11
RESTORE_SHOT_DONE_ALL_TARGET = 12
REMOVE_SHOT_DONE_ALL_TARGET = 13
REMOVE_SHOT_DONE_ALL_SLOT = 14
RESTORE_SHOT_DONE_ALL_SLOT = 15
UPDATE_RATING_ALL_SLOT = 16
```

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {“method”: “RemoteRoboTargetGetAnnotationListByRun”, “params”: {“RefGuidRun”:”30f7ea4b-a80e-4f3c-aaf8-b7049068d6bd”,”UID”:”a66120e5-6da7-4ccb-ab36-ad801e4c3fea”,”MAC”:”KxtTXZfYtTJre7IgvewTs9q/z4o=”}, “id”: 79}

🡐{“jsonrpc”: “2.0”, “result”: 0, “id”: 79}

🡐

{“Event”:”RemoteActionResult”,”Timestamp”:1652631731.35259,”Host”:”RC16”,”Inst”:1,”UID”:”a66120e5-6da7-4ccb-ab36-ad801e4c3fea”,”ActionResultInt”:4,”Motivo”:””,”ParamRet”:{“list”:[{ “guid”:”c0e6e2a8-23c4-4704-8e52-69ba15c6c3de”, “datetimestart”:1651318594, “code”:3, “note”:”User=admin;IsLocalAddress=False;RemoteIP=10.147.19.88;ObjUID=f94e820f-eff4-4b72-bf71-88e4fa32a10f;FileName=NGC4725_LIGHT_L_300s_BIN1_-12C_001_20220425_002621_686_GA_1087_OF_60_E.FIT”, “refguidrun”:”30f7ea4b-a80e-4f3c-aaf8-b7049068d6bd”, “refguidtarget”:”bc7387b5-2f4e-4c71-8055-e66e118383ec”, “targetname”:”NGC4725” },{ “guid”:”5d00fc52-099f-4594-828f-f7dd293cb2fe”, “datetimestart”:1651318596, “code”:3, “note”:”User=admin;IsLocalAddress=False;RemoteIP=10.147.19.88;ObjUID=038ae042-840f-4ef5-bc2e-b37c5a4858b8;FileName=NGC4725_LIGHT_L_300s_BIN1_-12C_002_20220425_003124_723_GA_1087_OF_60_E.FIT”, “refguidrun”:”30f7ea4b-a80e-4f3c-aaf8-b7049068d6bd”, “refguidtarget”:”bc7387b5-2f4e-4c71-8055-e66e118383ec”, “targetname”:”NGC4725” },{ “guid”:”7eccbe2f-1cee-4242-8090-03be62851659”, “datetimestart”:1651318598, “code”:3, “note”:”User=admin;IsLocalAddress=False;RemoteIP=10.147.19.88;ObjUID=b8560fd4-4b52-494°-8ed3-15dd215af8d2;FileName=NGC4725_LIGHT_L_300s_BIN1_-12C_003_20220425_003646_057_GA_1087_OF_60_E.FIT”, “refguidrun”:”30f7ea4b-a80e-4f3c-aaf8-b7049068d6bd”, “refguidtarget”:”bc7387b5-2f4e-4c71-8055-e66e118383ec”, “targetname”:”NGC4725” },{ “guid”:”bdeb5c55-9e9c-46fa-87cc-c0a017bc62a9”, “datetimestart”:1651318599, “code”:3, “note”:”User=admin;IsLocalAddress=False;RemoteIP=10.147.19.88;ObjUID=01addf13-9aab-4753-99e1-99ffd5a6fa5a;FileName=NGC4725_LIGHT_L_300s_BIN1_-12C_004_20220425_004557_868_GA_1087_OF_60_E.FIT”, “refguidrun”:”30f7ea4b-a80e-4f3c-aaf8-b7049068d6bd”, “refguidtarget”:”bc7387b5-2f4e-4c71-8055-e66e118383ec”, “targetname”:”NGC4725” }]}}

o) RemoteRoboTargetAddBaseSequence

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetAddBaseSequence</td></tr>
<tr><td>Description</td><td colspan="3">Add a Base Sequence (the sequence must exists in the default folder for Sequence Config in Voyager Folders) to a Profile</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>Guid</td><td>String</td><td>New UID to associate to the Base Sequence</td></tr>
<tr><td>Name</td><td>String</td><td>Base Sequence Name. You must use the</td></tr>
</table></td></tr>
</table>

			same Sequence filename with extension s2q
	FileName	String	Path and file name with extension of the Sequence
	ProfileName	String	Profile name within the sequence is associated (with extension .v2y)
	IsDefault	Boolean	True if is the default Base sequence for the Profile
	Status	Integer	Base Sequence Status 0=Enabled 1=Disabled
	Note	string	Text note associated to the Base Sequence
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		
Remote Action Result Parameters in ParamRet Object	ret	String	"DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetAddBaseSequence", "params": {"Guid":"b6b99e29-61c8-40ef-984f-ba5b719502b8","Name":"SequenzaBase_ColorTestAdvanced.s2q","FileName":"C:\\Users\\pegas\\OneDrive\\Documenti\\Voyager\\ConfigSequence\\SequenzaBase_ColorTestAdvanced.s2q","ProfileName":"ColorTestAdvanced.v2y","IsDefault":false,"Status":0,"Note":"","UID":"c74aaaaf-b710-4822-bbc6-b4d8886b45a4","MAC":"Cq/HX+UYeZWWIz9kxghI50g5/FY="}, "id": 10}

➔{"jsonrpc": "2.0", "result": 0, "id": 10}

➔

{"Event":"RemoteActionResult","Timestamp":1652641250.91671,"Host":"ORIONE","Inst":1,"UID":"c74aaaaf-b710-4822-bbc6-b4d8886b45a4","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

p) RemoteRoboTargetUpdateBaseSequence

Method	`RemoteRoboTargetUpdateBaseSequence`		
Description	Update a Base Sequence already stored		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidBaseSequence	String	UID of Base Sequence

	IsDefault	Boolean	True if is the default Base sequence for the Profile
	Status	Integer	Base Sequence Status 0=Enabled 1=Disabled
	Note	string	Text note associated to the Base Sequence
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		
Remote Action Result Parameters in ParamRet Object	ret	String	"DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetUpdateBaseSequence", "params": {"RefGuidBaseSequence":"b6b99e29-61c8-40ef-984f-ba5b719502b8","IsDefault":false,"Status":0,"Note":"rrr","UID":"4d094695-4e3f-4537-bfea-c4e7c8b95612","MAC":"hnWWGtk7Qr2o0/y6Qib1X5RO+g8="}, "id": 15}

←{"jsonrpc": "2.0", "result": 0, "id": 15}

←
{"Event":"RemoteActionResult","Timestamp":1652642002.29062,"Host":"ORIONE","Inst":1,"UID":"4d094695-4e3f-4537-bfea-c4e7c8b95612","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

q) RemoteRoboTargetRemoveBaseSequence

Method	`RemoteRoboTargetRemoveBaseSequence`		
Description	Remove a Base Sequence already stored, attention all the Target referring to this Base Sequence will become Orphan and cannot be used for scheduling until you will have fixed the new Base Sequence associated to Target		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidBaseSequence	String	UID of Base Sequence
	MAC	String	Create a concatenated string with RoboTarget Shared secret +

			SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		
Remote Action Result Parameters in ParamRet Object	ret	String	"DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetRemoveBaseSequence", "params": {"RefGuidBaseSequence":"b6b99e29-61c8-40ef-984f-ba5b719502b8","UID":"dbca3899-51b7-4848-bdad-2bef370f149d","MAC":"8FMMUFFRquO0uj3uJRinriHEUh8="}, "id": 12}

➔{"jsonrpc": "2.0", "result": 0, "id": 12}

➔
{"Event":"RemoteActionResult","Timestamp":1652858769.98685,"Host":"ORIONE","Inst":1,"UID":"dbca3899-51b7-4848-bdad-2bef370f149d","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

r) RemoteRoboTargetAddSet

Method	`RemoteRoboTargetAddSet`		
Description	Add a Set to a Profile		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	Guid	String	New UID to associate to the Set
	Name	String	Set Name
	ProfileName	String	Profile name within the Set is associated (with extension .v2y)
	IsDefault	Boolean	True if is the default Set for the Profile
	Tag	String	Tag of Set
	Status	Integer	Set Status 0=Enabled 1=Disabled
	Note	string	Text note associated to the Set
	MAC	String	Create a concatenated string with

			RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret | String | "DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetAddSet", "params": {"Guid":"28f89109-9010-4283-b191-7b3d1665e0e1","Name":"Pippolo","ProfileName":"TestFlatNoMount.v2y","IsDefault":false,"Status":0,"Note":"","UID":"39b61128-8°6a-41b8-b5d5-f35dece9cd0c","MAC":"0bNFLTMSnFwXehc2vO3bQ+FLg8A="}, "id": 9}

➔{"jsonrpc": "2.0", "result": 0, "id": 9}

➔
{"Event":"RemoteActionResult","Timestamp":1652908140.20808,"Host":"ORIONE","Inst":1,"UID":"39b61128-8°6a-41b8-b5d5-f35dece9cd0c","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

s) RemoteRoboTargetUpdateSet

Method	`RemoteRoboTargetUpdateSet`
Description	Update a Set

Params

UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
RefGuidSet	String	UID of the Set to Update
Name	String	Set Name
Status	Integer	Set Status 0=Enabled 1=Disabled
Tag	String	Tag of Set
Note	string	Text note associated to the Set
MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string

			received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret | String | "DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetUpdateSet", "params": {"RefGuidSet":"28f89109-9010-4283-b191-7b3d1665e0e1","Name":"Pippolone","Status":0,"Note":"","UID":"518d2b62-a3be-40fb-b269-41°7e36cc089","MAC":"jGWUA3aapwfPPVzTJQI3Ak/EerM="}, "id": 10}

←{"jsonrpc": "2.0", "result": 0, "id": 10}

←
{"Event":"RemoteActionResult","Timestamp":1652908463.05184,"Host":"ORIONE","Inst":1,"UID":"518d2b62-a3be-40fb-b269-41°7e36cc089","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

t) RemoteRoboTargetRemoveSet

Method	`RemoteRoboTargetRemoveSet`
Description	Remove a Set already stored, attention all the Target, shot, and generally data referring to this Set will be deleted

Params

UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
RefGuidSet	String	UID of Set
MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret | String | "DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetRemoveSet", "params": {"RefGuidSet":"28f89109-9010-4283-b191-7b3d1665e0e1","UID":"8406b027-22f0-459e-af23-c2b729679b09","MAC":"rP0zXK20cgnXdmc3kkgci742LuI="}, "id": 11}

🡐{"jsonrpc": "2.0", "result": 0, "id": 11}

🡐
{"Event":"RemoteActionResult","Timestamp":1652908644.33225,"Host":"ORIONE","Inst":1,"UID":"8406b027-22f0-459e-af23-c2b729679b09","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

u) RemoteRoboTargetAddTarget

Method	`RemoteRoboTargetAddTarget`
Description	Add a Target to a Set
Params	UID | String | Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated GuidTarget | String | New UID to associate to the Target RefGuidSet | String | UID of Set RefGuidBaseSequence | String | UID of Base Sequence TargetName | String | Name of Target Tag | String | Tag of Target RAJ2000 | Numeric | RA coord is J2000 format expressed in hours DECJ2000 | Numeric | DEC coord is J2000 format expressed in degrees PA | numeric | PA expressed in degrees of the Target, will be used for rotator if configured in base sequence. Sky PA or Mechanical PA depends on how is configured the Base Sequence DateCreation | datetime | Date time of creation target

			to store in DB (will be used from the scheduler to order the targets)
	Status	integer	0 = Enabled 1 = Disabled (will be not used for scheduling)
	StatusOp	integer	Target operative status: -1 = Unknow 0 = Idle 1 = Running 2 = Finished 3 = Ephemeris not calculated 4 = Expired
	Note	string	Text note about the target
	IsRepeat	boolean	True = the shot group configured for this target will be repeated in sequence
	Repeat	integer	Number of repeat of shot group defined for this target (see the previous parameter)
	IsFinishActualExposure	Boolean	True = finish the actual exposure in case of time end for sequence expired
	IsCoolSetPoint	Boolean	Override in the sequence the cooling set point temperature if already enabled the cooling in the base sequence (not switch on cooling)
	CoolSetPoint	integer	Cooling temperature
	IsWaitShot	Boolean	Override in the sequence the wait time between shot if already enabled in base sequence
	WaitShot	integer	Time in seconds to wait between shot
	IsGuideTime	Boolean	Override in the sequence the guiding time exposure if already enabled in base sequence
	GuideTime	Numeric	Exposure time seconds for guiding shot
	Priority	integer	Target priority to use for scheduler target ordering 0 = Very Low 1 = Low 2 = Normal 3 = High 4 = First
	C_ID	string	UID of constraints set
	C_Mask	string	List of chars that report the enabled constraints for this target

			A=Position Angle of Target B=Min Altitude of Target C=Min SQM D=HA Start time E=HA End Time F=Start Date G=End Date H=Min allowed time start J=Max allowed time start K=Moon Down L=Moon Phase min M=Moon Phase max N=Moon Distance O=HFD Sub Max P=Max Sequence time for night Q=Air mass min R=Air mass max S=Moon Distance Lorentzian T=Max Sequence Time U=One Shot V=Preset Time Interval
	C_AltMin	numeric	Minimum Altitude of Target in Degrees
	C_SqmMin	numeric	Min SQM allowed for target running, you must have an SQM control attached to Voyager
	C_HAStart	numeric	Minimum HA allowed for target
	C_HAEnd	numeric	Maxim HA allowed for target
	C_DateStart	datetime	Minimum date useful to start a target session
	C_DateEnd	Datetime	Maxim date useful to start a target session
	C_TimeStart	datetime	Minimum time for start a target session
	C_TimeEnd	datetime	Maximum time for start a target session
	C_MoonDown	boolean	True if the target must be scheduled only if moon is under the altitude set like moon down condition (see RoboTarget settings)
	C_MoonPhaseMin	numeric	Min Phase valid for scheduling target
	C_MoonPhaseMax	numeric	Max Phase valid for scheduling target
	C_MoonDistanceDegree	numeric	Minimum moon distance in degree to scheduling a target
	C_HFDMeanLimit	numeric	Maximum HFD value of a shot (calculated on all the field)

			valid for continue the target session running
	C_MaxTimeForDay	numeric	Max time global in minutes allowed for target session in 24 hours
	C_AirMassMin	Numeric	Min Airmass to scheduling target
	C_AirMassMax	Numeric	Max airmass to scheduling target
	C_MoonDistanceLorentzian	integer	Profile of Lorentzian moon avoidance to use BROAD_BAND = 0 NARROW_BAND = 1 FREE = 2
	C_MaxTime	integer	Max time in minutes allowed for target session
	C_OSDateStart	datetime	Date for oneshot target start
	C_OSTimeStart	datetime	Time for oneshot target start
	C_OSEarly	Integer	Minute to start early for oneshot target
	C_PINTEarly	Integer	Minute to start early for preset time interval target
	C_PINTReset	boolean	Reset Progress at each sequence run
	C_PINTIntervals	array	JSON object Array of interval in JSON format. See structure of Array in paragraph 7
	C_Mask2	String	List of string to define secondary specialized constraints for target L01=look at C_L01 M01=look at C_M01 N01=look at C_N01 S01=look at C_S01
	C_L01	Boolean	And Moon Up for the Moon Phase min constraints (L)
	C_M01	Boolean	Or Moon Down for Moon phase max constraint (M)
	C_N01	boolean	Or moon Down for Moon Distance constraint (N)
	C_S01	boolean	Or moon Down for Lorentzian Moon Avoidance constraint (S)
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC

			command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
	Token	String	Reserved OpenSkyGems
	TKey	String	Search Key for Dynamic Target (Voyager CTNAME Field for Asteroid as for dynamic target sample CSV received). Match is only with this field !
	TName	String	Designation Name of the Dynamic Object. Just an info, will not be used for search but is useful for who use the RoboTarget Manager to have in TName field of the Target a better name than the TKey string
	TType	Integer	0=DSO/Default 1=Comet , 2=Asteroid, 3=Planet, 4=DynaSearch . Using value of 1,2,3,4 the target is declared by Voyager Dynamic and Voyager will use the TKey for search the object in RoboOrbits and calculate the new cords RA/DEC when requested. MUST VALORIZED ever !
	IsDynamicPointingOverride	boolean	True if you want override the dynamic pointing mode of the base sequence for the target. Otherwise Voyager will use the Dynamic Pointing mode configured in the base sequence
	DynamicPointingOverride	integer	If IsDynamicPointingOverride is true you can define when Voyager will calculate with RoboOrbits then RA/DEC of the target. This Dynamic Pointing mode to use to override the base sequence. Values are 0=Begin of Sequence, 1= Each Goto in the Sequence, 2= Each X Seconds
	DynEachX_Seconds	integer	If you have defined the Each X Seconds DynamicPointingOverride you can override the number of seconds for the interval
	DynEachX_Realign	boolean	If you have defined the Each X Seconds DynamicPointingOverride you

			can override if Voyager will realign the target as soon as possible when the X seconds is passed by
	DynEachX_NoPlateSolve	boolean	If you have defined the Each X Seconds DynamicPointingOverride and select to Realign you can define if use the plate solving for the pointing during the realign or just do a realign goto
	IsOffsetRF	boolean	True = Enable the adding of an offset in steps to the final position of the focus, this will override the base sequence and the overall offset in RoboFire Settings (Works only for RoboFire)
	OffsetRF	integer	Steps (positive or negative) to add to the final focus. Overrides only if IsOffsetRF is true

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	

ret	String	“DONE” if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetAddTarget", "params": {"GuidTarget":"76878ebc-55d0-4ffd-8298-a726d1625c2d","RefGuidSet":"5482d20e-2304-41d1-8d2b-32adc2c314bc","RefGuidBaseSequence":"90ae5721-a248-4159-ad74-56e13cf26141","TargetName":"vdB 1","RAJ2000":0.176666666666667,"DECJ2000":58.7666666666667,"PA":120,"DateCreation":1653120454.91103,"Status":0,"StatusOp":0,"Note":"Text Note","IsRepeat":true,"Repeat":4,"Priority":2,"C_ID":"c3d81012-ddca-4617-924e-c5a61f49240a","C_Mask":"ABCDEFGHJKLMNOPQR","C_AltMin":22,"C_SqmMin":21.3,"C_HAStart":-1,"C_HAEnd":2,"C_DateStart":1652832000,"C_DateEnd":1653177600,"C_TimeStart":1653094923,"C_TimeEnd":1653105906,"C_MoonDown":true,"C_MoonPhaseMin":10,"C_MoonPhaseMax":90,"C_MoonDistanceDegree":23,"C_HFDMeanLimit":5.3,"C_MaxTimeForDay":120,"C_AirmassMin":1.234,"C_AirmassMax":2.345,"C_MoonDistanceLorentzian":0,"C_Mask2":"L01M01N01S01","C_L01":true,"C_M01":true,"C_N01":true, ,"C_S01":true,"MAC":"yW3Y8aRExk3Yf2qA/JTAMruu4Dc=","UID":"8766074d-f415-4cad-826b-b0e2a5ae40b7"}, "id": 12}

➔{"jsonrpc": "2.0", "result": 0, "id": 12}

←

{"Event":"RemoteActionResult","Timestamp":1653120455.07401,"Host":"ORIONE","Inst":1,"UID":"8766074d-f415-4cad-826b-b0e2a5ae40b7","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

ALWAYS CHECK the param ret = DONE in the RemoteActionResult ParameRet, a ret different from DONE is an Error.

v) RemoteRoboTargetUpdateTarget

Method	RemoteRoboTargetUpdateTarget		
Description	Update and exists Target		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidTarget	String	UID to associate to the Target
	RefGuidSet	String	UID of Set
	RefGuidBaseSequence	String	UID of Base Sequence
	TargetName	String	Name of Target
	Tag	String	Tag of Target
	RAJ2000	Numeric	RA coord is J2000 format expressed in hours
	DECJ2000	Numeric	DEC coord is J2000 format expressed in degrees
	PA	numeric	PA expressed in degrees of the Target, will be used for rotator if configured in base sequence. Sky PA or Mechanical PA depends on how is configured the Base Sequence
	DateCreation	datetime	Date time of creation target to store in DB (will be used from the scheduler to order the targets)
	Status	integer	0 = Enabled 1 = Disabled (will be not used for scheduling)
	StatusOp	integer	Target operative status: -1 = Unknow 0 = Idle 1 = Running 2 = Finished 3 = Ephemeris not calculated 4 = Expired
	Note	string	Text note about the target
	IsRepeat	boolean	True = the shot group configured for this target will be repeated in sequence

	Repeat	integer	Number of repeat of shot group defined for this target (see the previous parameter)
	IsCoolSetPoint	Boolean	Override in the sequence the cooling set point temperature if already enabled the cooling in the base sequence (not switch on cooling)
	CoolSetPoint	integer	Cooling temperature
	IsWaitShot	Boolean	Override in the sequence the wait time between shot if already enabled in base sequence
	WaitShot	integer	Time in seconds to wait between shot
	IsGuideTime	Boolean	Override in the sequence the guiding time exposure if already enabled in base sequence
	GuideTime	Numeric	Exposure time seconds for guiding shot
	IsFinishActualExposure	Boolean	True = finish the actual exposure in case of time end for sequence expired
	Priority	integer	Target priority to use for scheduler target ordering 0 = Very Low 1 = Low 2 = Normal 3 = High 4 = First
	C_ID	string	UID of constraints set (optional field)
	C_Mask	string	List of chars that report the enabled constraints for this target A=Position Angle of Target B=Min Altitude of Target C=Min SQM D=HA Start time E=HA End Time F=Start Date G=End Date H=Min allowed time start J=Max allowed time start K=Moon Down L=Moon Phase min M=Moon Phase max N=Moon Distance O=HFD Sub Max P=Max Sequence time for night

			Q=Air mass min R=Air mass max S=Moon Distance Lorentzian T=Max Sequence Time U=One Shot V=Preset Time Interval
	C_AltMin	numeric	Minimum Altitude of Target in Degrees
	C_SqmMin	numeric	Min SQM allowed for target running, you must have an SQM control attached to Voyager
	C_HAStart	numeric	Minimum HA allowed for target
	C_HAEnd	numeric	Maxim HA allowed for target
	C_DateStart	datetime	Minimum date useful to start a target session
	C_DateEnd	Datetime	Maxim date useful to start a target session
	C_TimeStart	datetime	Minimum time for start a target session
	C_TimeEnd	datetime	Maximum time for start a target session
	C_MoonDown	boolean	True if the target must be scheduled only if moon is under the altitude set like moon down condition (see RoboTarget settings)
	C_MoonPhaseMin	numeric	Min Phase valid for scheduling target
	C_MoonPhaseMax	numeric	Max Phase valid for scheduling target
	C_MoonDistanceDegree	numeric	Minimum moon distance in degree to scheduling a target
	C_HFDMeanLimit	numeric	Maximum HFD value of a shot (calculated on all the field) valid for continue the target session running
	C_MaxTimeForDay	numeric	Max time global in minutes allowed for target session in 24 hours
	C_AirMassMin	Numeric	Min Airmass to scheduling target
	C_AirMassMax	Numeric	Max airmass to scheduling target

	C_MoonDistanceLorentzian	integer	Profile of Lorentzian moon avoidance to use
			
			`BROAD_BAND = 0`
			`NARROW_BAND = 1`
			`FREE = 2`

	C_MaxTime	integer	Max time in minutes allowed for target session
	C_OSDateStart	datetime	Date for oneshot target start
	C_OSTimeStart	datetime	Time for oneshot target start
	C_OSEarly	Integer	Minute to start early for oneshot target
	C_PINTEarly	Integer	Minute to start early for preset time interval target
	C_PINTReset	boolean	Reset Progress at each sequence run
	C_PINTIntervals	array	JSON object Array of interval in JSON format. See structure of Array in paragraph 7
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
	C_Mask2	String	List of string to define secondary specialized constraints for target L01=look at C_L01 M01=look at C_M01 N01=look at C_N01 S01=look at C_S01
	C_L01	Boolean	And Moon Up for the Moon Phase min constraints (L)
	C_M01	Boolean	Or Moon Down for Moon phase max constraint (M)
	C_N01	boolean	Or moon Down for Moon Distance constraint (N)
	C_S01	boolean	Or moon Down for Lorentzian Moon Avoidance constraint (S)
	Token	string	Reserved to OpenSkyGems
	TKey	String	Search Key for Dynamic Target (Voyager CTNAME Field for Asteroid as for dynamic target sample CSV received). Match is only with this field !
	TName	String	Designation Name of the Dynamic Object. Just an info, will not be used for search

			but is useful for who use the RoboTarget Manager to have in TName field of the Target a better name than the TKey string
	TType	Integer	0=DSO/Default , 1=Comet , 2=Asteroid, 3=Planet, 4=DynaSearch . Using value of 1,2,3,4 the target is declared by Voyager Dynamic and Voyager will use the TKey for search the object in RoboOrbits and calculate the new cords RA/DEC when requested. MUST VALORIZED ever !
	IsDynamicPointingOverride	boolean	True if you want override the dynamic pointing mode of the base sequence for the target. Otherwise Voyager will use the Dynamic Pointing mode configured in the base sequence
	DynamicPointingOverride	integer	If IsDynamicPointingOverride is true you can define when Voyager will calculate with RoboOrbits then RA/DEC of the target. This Dynamic Pointing mode to use to override the base sequence. Values are 0=Begin of Sequence, 1= Each Goto in the Sequence, 2= Each X Seconds
	DynEachX_Seconds	integer	If you have defined the Each X Seconds DynamicPointingOverride you can override the number of seconds for the interval
	DynEachX_Realign	boolean	If you have defined the Each X Seconds DynamicPointingOverride you can override if Voyager will realign the target as soon as possible when the X seconds is passed by
		boolean	If you have defined the Each X Seconds DynamicPointingOverride and select to Realign you can define if use the plate solving for the pointing during the

			realign or just do a realign goto
	IsOffsetRF	boolean	True = Enable the adding of an offset in steps to the final position of the focus, this will override the base sequence and the overall offset in RoboFire Settings (Works only for RoboFire)
	OffsetRF	integer	Steps (positive or negative) to add to the final focus. Overrides only if IsOffsetRF is true

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret | String | "DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetUpdateTarget", "params": {"RefGuidTarget":"76878ebc-55d0-4ffd-8298-a726d1625c2d","RefGuidSet":"5482d20e-2304-41d1-8d2b-32adc2c314bc","RefGuidBaseSequence":"90ae5721-a248-4159-ad74-56e13cf26141","TargetName":"vdB 1","RAJ2000":0.176666666666667,"DECJ2000":58.7666666666667,"PA":120,"DateCreation":1653129457.99933,"Status":0,"StatusOp":0,"Note":"Text Note 2","IsRepeat":true,"Repeat":4,"Priority":2,"C_ID":"","C_Mask":"ABCDEFGHJKLMNOPQR","C_AltMin":22,"C_SqmMin":21.3,"C_HAStart":-1,"C_HAEnd":2,"C_DateStart":1652832000,"C_DateEnd":1653177600,"C_TimeStart":3723,"C_TimeEnd":14706,"C_MoonDown":true,"C_MoonPhaseMin":10,"C_MoonPhaseMax":90,"C_MoonDistanceDegree":23,"C_HFDMeanLimit":0,"C_MaxTimeForDay":120,"C_AirmassMin":1.234,"C_AirmassMax":2.345,"C_MoonDistanceLorentzian":0,"C_Mask2":"L01M01N01S01","C_L01":true,"C_M01":true,"C_N01":true,,"C_S01":true,"MAC":"sJEsCUH9aNM4GlaC6IPCSR6lohg=","UID":"7bba69e8-5928-46d3-99be-7db26d0310b8"}, "id": 21}

➔{"jsonrpc": "2.0", "result": 0, "id": 21}

➔

{"Event":"RemoteActionResult","Timestamp":1653129458.25283,"Host":"ORIONE","Inst":1,"UID":"7bba69e8-5928-46d3-99be-7db26d0310b8","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

ALWAYS CHECK the param ret = DONE in the RemoteActionResult ParameRet, a ret different from DONE is an Error.

w) RemoteRoboTargetRemoveTarget

Method	RemoteRoboTargetRemoveTarget
Description	Remove a Target already stored, attention all the shot, and generally data referring to this Target will be deleted
Params	UID — String — Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated RefGuidTarget — String — UID of Target MAC — String — Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.
Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret — String — "DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetRemoveTarget", "params": {"RefGuidTarget":"8ee22f98-11f8-45b8-931d-ae5f373df143","UID":"1ecd5289-51b4-423d-885a-5c6a1e70572e","MAC":"GjD+5GHvFoeSCWpRoz2ir+lJKgs="}, "id": 24}

←{"jsonrpc": "2.0", "result": 0, "id": 11}

←
{"Event":"RemoteActionResult","Timestamp":1653129736.88071,"Host":"ORIONE","Inst":1,"UID":"1ecd5289-51b4-423d-885a-5c6a1e70572e","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

x) RemoteRoboTargetGetConfigDataShot

Method	RemoteRoboTargetGetConfigDataShot
Description	Return information about Shot configuration usable for the profile in parameter
Params	UID — String — Unique identifier of the Action to abort. Use a Guide Window identifier or a

			unique key string generated
	ProfileName	String	Name of profile file with extension . Empty to retrieve data for all profile available in the default profile folder of Voyager
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*

Remote Action Result Parameters in ParamRet Object

List — object — Array of Shot configuration possible settings

guid	String	UID of Shot Configuration
name	string	Profile file name with extension
isactive	boolean	True if the profile is the actual selected in Voyager
sensortype	integer	Sensor type 0= Monochrome 1=Color 2=DSLR
iscmos	boolean	True if is a CMOS sensor
filters	object	Filters configuration for the profile
Filternum	integer	Number of filters configured
Filter<x>_Name	string	Name of filter <x>
Filter<x>_MagMin	integer	Min magnitude of focus star for the filter
Filter<x>_MagMax	integer	Max magnitude of

					focus star for the filter
			Filter<x>_Offset	integer	Filter offset
			readoutmode	object	Readoutmodes available for the profile
			ReadoutNum	integer	Number of readoutmodes configured
			Readout<x>_Name	string	Name of readoutmode
			Readout<x>_Index	integer	Index of readoutmode to use in the camera shot command
			speed	object	Speed available for the profile
			SpeedNum	integer	Number of speeds configures
			Speed<x>_Name	string	Name of speed
			Speed<x>_Index	integer	Index of speed to use in camera shot command
			targetname	string	Target name if available

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetConfigDataShot", "params": {"ProfileName":"","UID":"f2fe90ef-889a-43d0-9ac9-308a51051be9","MAC":"SAkVmuzbcWdzq1+OnUeDE4YPF/A="}, "id": 16}

←{"jsonrpc": "2.0", "result": 0, "id": 16}

←

{"Event":"RemoteActionResult","Timestamp":1653129442.32527,"Host":"ORIONE","Inst":1,"UID":"f2fe90ef-889a-43d0-9ac9-308a51051be9","ActionResultInt":4,"Motivo":"","ParamRet":{"list":[{"guid":"cf2e80dc-d971-474d-a075-fb5a9d81b1a3", "name":"ColorTestAdvanced.v2y", "isactive":false, "sensortype":1, "iscmos":true, "filters":{"FilterNum":0}, "readoutmode":{"ReadoutNum":1,"Readout1_Name":"16 bit","Readout1_Index":0}, "speed":{"SpeedNum":1,"Speed1_Name":"Default","Speed1_Index":0}},{"guid":"c2b8c7bb-0fc3-42c1-914f-4af49f94ef8d", "name":"temp.v2y", "isactive":false, "sensortype":0, "iscmos":true, "filters":{"FilterNum":7,"Filter1_Name":"L","Filter1_MagMin":3,"Filter1_MagMax":7,"Filter1_Offset":0, "Filter2_Name":"r","Filter2_MagMin":3,"Filter2_MagMax":7,"Filter2_Offset":0,"Filter3_Name":"g","Filter3_MagMin":3,"Filter3_MagMax":7,"Filter3_Offset":0,"Filter4_Name":"b","Filter4_MagMin":3,"Filter4

_MagMax":7,"Filter4_Offset":0,"Filter5_Name":"ha","Filter5_MagMin":2,"Filter5_MagMax":7,"Filter5_Offset":0,"Filter6_Name":"o3","Filter6_MagMin":2,"Filter6_MagMax":7,"Filter6_Offset":0,"Filter7_Name":"s2","Filter7_MagMin":2,"Filter7_MagMax":7,"Filter7_Offset":0},
"readoutmode":{"ReadoutNum":1,"Readout1_Name":"Default","Readout1_Index":0},
"speed":{"SpeedNum":1,"Speed1_Name":"Default","Speed1_Index":0}},{"guid":"3a2b82b3-621f-4e84-83b2-a1befb154af5", "name":"TestFlatNoMount.v2y", "isactive":true, "sensortype":0, "iscmos":false,
"filters":{"FilterNum":8,"Filter1_Name":"L","Filter1_MagMin":4,"Filter1_MagMax":7,"Filter1_Offset":0,"Filter2_Name":"R","Filter2_MagMin":4,"Filter2_MagMax":7,"Filter2_Offset":0,"Filter3_Name":"G","Filter3_MagMin":4,"Filter3_MagMax":7,"Filter3_Offset":0,"Filter4_Name":"B","Filter4_MagMin":4,"Filter4_MagMax":7,"Filter4_Offset":0,"Filter5_Name":"HA","Filter5_MagMin":4,"Filter5_MagMax":7,"Filter5_Offset":0,"Filter6_Name":"OIII","Filter6_MagMin":4,"Filter6_MagMax":7,"Filter6_Offset":0,"Filter7_Name":"SII","Filter7_MagMin":4,"Filter7_MagMax":7,"Filter7_Offset":0,"Filter8_Name":"CLEAR","Filter8_MagMin":4,"Filter8_MagMax":7,"Filter8_Offset":0},
"readoutmode":{"ReadoutNum":1,"Readout1_Name":"16 bit","Readout1_Index":0},
"speed":{"SpeedNum":1,"Speed1_Name":"Default","Speed1_Index":0}}]}}

y) RemoteRoboTargetAddShot

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetAddShot</td></tr>
<tr><td>Description</td><td colspan="3">Add a Shot configuration (Slot) to a Target</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort.
Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>GuidShot</td><td>String</td><td>New UID to associate to the Shot</td></tr>
<tr><td>RefGuidTarget</td><td>String</td><td>UID of Target</td></tr>
<tr><td>Label</td><td>String</td><td>Suffix to append to the file name (optional)</td></tr>
<tr><td>FilterIndex</td><td>integer</td><td>Index of filter how configured for the profile (0 is always allow edfor default filter). Get the list with command RemoteRoboTargetGetConfigDataShot</td></tr>
<tr><td>Num</td><td>Integer</td><td>How many shot for this slot</td></tr>
<tr><td>Bin</td><td>integer</td><td>Binning to use</td></tr>
<tr><td>ReadoutMode</td><td>integer</td><td>Index of readoutmode to use (0 is always allowed fro default readoutmode). Get the list with command RemoteRoboTargetGetConfigDataShot</td></tr>
<tr><td>Type</td><td>integer</td><td>Image type:
0=Light
1=Bias
2=Dark
3=Flat</td></tr>
<tr><td>Speed</td><td>integer</td><td>Index of Speed to use (0 is always allowed fro default Speed). Get the list with command RemoteRoboTargetGetConfigDataShot</td></tr>
<tr><td>Gain</td><td>Integer</td><td>Dedicated to CMOS, gain to use for shot</td></tr>
</table>
</td></tr>
</table>

			(work only if the camera is declared CMOS in Voyager). Use 0 otherwise
	Offset	Integer	Dedicated to CMOS, offset to use for shot (work only if the camera is declared CMOS in Voyager). Use 0 otherwise
	Exposure	Numeric	Time exposure expressed in seconds
	Order	Integer	Order of the slot in the sequence. Order of shot.
	Done	boolean	Not used
	Enabled	boolean	True if the Slot is enabled. If not enable the slot will be not used to create the sequence
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		
Remote Action Result Parameters in ParamRet Object	ret	String	"DONE" if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetAddShot", "params": {"GuidShot":"94bff9f0-73fe-4913-83b7-8f2e1ace1943","RefGuidTarget":"76878ebc-55d0-4ffd-8298-a726d1625c2d","Label":"Suffix","FilterIndex":0,"Num":10,"Bin":1,"ReadoutMode":0,"Type":0,"Speed":0,"Gain":0,"Offset":0,"Exposure":300,"Order":-1,"Done":false,"Enabled":true,"MAC":"lwMbdJ4tIermJRXScBkcuCE8QuM=","UID":"eef5f84b-f2e3-41f6-ad0a-6cd422b52e95"}, "id": 27}

➔{"jsonrpc": "2.0", "result": 0, "id": 27}

➔
{"Event":"RemoteActionResult","Timestamp":1653131207.63764,"Host":"ORIONE","Inst":1,"UID":"eef5f84b-f2e3-41f6-ad0a-6cd422b52e95","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

z) RemoteRoboTargetUpdateShot

Method	`RemoteRoboTargetUpdateShot`
Description	Update a Shot (Slot) configuration

Params

UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
RedGuidShot	String	UID to associate to the Shot
RefGuidTarget	String	UID of Target
Label	String	Suffix to append to the file name (optional)
FilterIndex	integer	Index of filter how configured for the profile (0 is always allow edfor default filter). Get the list with command RemoteRoboTargetGetConfigDataShot
Num	Integer	How many shot for this slot
Bin	integer	Binning to use
ReadoutMode	integer	Index of readoutmode to use (0 is always allowed fro default readoutmode). Get the list with command RemoteRoboTargetGetConfigDataShot
Type	integer	Image type: 0=Light 1=Bias 2=Dark 3=Flat
Speed	integer	Index of Speed to use (0 is always allowed fro default Speed). Get the list with command RemoteRoboTargetGetConfigDataShot
Gain	Integer	Dedicated to CMOS, gain to use for shot (work only if the camera is declared CMOS in Voyager). Use 0 otherwise
Offset	Integer	Dedicated to CMOS, offset to use for shot (work only if the camera is declared CMOS in Voyager). Use 0 otherwise
Exposure	Numeric	Time exposure expressed in seconds
Order	Integer	Order of the slot in the sequence. Order of shot.
Done	boolean	Not used
Enabled	boolean	True if the Slot is enabled. If not enable the slot will be not used to create the sequence
MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.

Result

Integer(0)

License Required

Advanced, Full

Remote Action Result Parameters

ret	String	“DONE” if ok otherwise is an error

in ParamRet Object	

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetUpdateShot", "params": {"RefGuidShot":"94bff9f0-73fe-4913-83b7-8f2e1ace1943","RefGuidTarget":"76878ebc-55d0-4ffd-8298-a726d1625c2d","Label":"Suffix3","FilterIndex":0,"Num":10,"Bin":1,"ReadoutMode":0,"Type":0,"Speed":0,"Gain":0,"Offset":0,"Exposure":300,"Order":-1,"Done":false,"Enabled":true,"MAC":"f6DetyCGAMSOu18+Ecbf7K7Qmhs=","UID":"239ad356-5093-43f7-9dfd-7440415013bd"}, "id": 29}

➔{"jsonrpc": "2.0", "result": 0, "id": 29}

➔
{"Event":"RemoteActionResult","Timestamp":1653131854.95342,"Host":"ORIONE","Inst":1,"UID":"239ad356-5093-43f7-9dfd-7440415013bd","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

aa) RemoteRoboTargetRemoveShot

Method	`RemoteRoboTargetRemoveShot`
Description	Remove a Shot (Slot) already stored, attention all the data referring to this will be removed
Params	UID — String — Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated RefGuidShot — String — UID of Shot (Slot) MAC — String — Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.
Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	ret — String — “DONE” if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetRemoveShot", "params": {"RefGuidShot":"94bff9f0-73fe-4913-83b7-8f2e1ace1943","UID":"21f97a30-97c7-4078-8f30-1ccd38ad4f37","MAC":"ijB5tVBMx5UmBGp54Gvc5NLH/xQ="}, "id": 31}

🡐{"jsonrpc": "2.0", "result": 0, "id": 31}

🡐
{"Event":"RemoteActionResult","Timestamp":1653131995.13992,"Host":"ORIONE","Inst":1,"UID":"21f97a30-97c7-4078-8f30-1ccd38ad4f37","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

bb) RemoteRoboTargetMoveShot

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetMoveShot</td></tr>
<tr><td>Description</td><td colspan="3">Change order of a Shot (Slot)</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidShot</td><td>String</td><td>UID of Shot (Slot)</td></tr>
<tr><td>MoveType</td><td>integer</td><td>How to move the Shot (Slot)
0=First
1=Up
2=Down
3=Last</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3">
<table>
<tr><td>ret</td><td>String</td><td>“DONE” if ok otherwise is an error</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetMoveShot", "params": {"RefGuidShot":"14dca058-f03e-46e1-8a00-1d9f55c061c4","MoveType":1,"MAC":"2Sh43HRTnRslS7oBT7q6oGVkUGY=","UID":"0d0ad256-f5ee-45e9-b60f-0cff65ce4f61"}, "id": 33}

➔{"jsonrpc": "2.0", "result": 0, "id": 33}

➔
{"Event":"RemoteActionResult","Timestamp":1653132344.9702,"Host":"ORIONE","Inst":1,"UID":"0d0ad256-f5ee-45e9-b60f-0cff65ce4f61","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

cc)RemoteRoboTargetDisableAllTargetsInSet

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetDisableAllTargetInSet</td></tr>
<tr><td>Description</td><td colspan="3">Disable all the Targets in a Set</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidSet</td><td>String</td><td>UID of Set</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3">
<table>
<tr><td>ret</td><td>String</td><td>“DONE” if ok otherwise is an error</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetDisableAllTargetsInSet", "params": {"RefGuidSet":"5482d20e-2304-41d1-8d2b-32adc2c314bc","UID":"bc828136-a91b-4ca4-8609-be420838c9f2","MAC":"gBlP/sLfihBWzBnj1alBfZuJ46A="}, "id": 36}

➔{"jsonrpc": "2.0", "result": 0, "id": 36}

←

{"Event":"RemoteActionResult","Timestamp":1653132623.75549,"Host":"ORIONE","Inst":1,"UID":"bc828136-a91b-4ca4-8609-be420838c9f2","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

dd) RemoteRoboTargetEnableAllTargetsInSet

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetEnableAllTargetInSet</td></tr>
<tr><td>Description</td><td colspan="3">Enable all the Targets in a Set</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidSet</td><td>String</td><td>UID of Set</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
</table></td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3"><table>
<tr><td>ret</td><td>String</td><td>“DONE” if ok otherwise is an error</td></tr>
</table></td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetEnableAllTargetsInSet", "params": {"RefGuidSet":"5482d20e-2304-41d1-8d2b-32adc2c314bc","UID":"bc828136-a91b-4ca4-8609-be420838c9f2","MAC":"gBlP/sLfihBWzBnj1alBfZuJ46A="}, "id": 36}

←{"jsonrpc": "2.0", "result": 0, "id": 36}

←

{"Event":"RemoteActionResult","Timestamp":1653132623.75549,"Host":"ORIONE","Inst":1,"UID":"bc828136-a91b-4ca4-8609-be420838c9f2","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

ee) RemoteRoboTargetMoveCopyTarget

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetMoveCopyTarget</td></tr>
<tr><td>Description</td><td colspan="3">Move or Copy Target to another Set/Profile</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidTarget</td><td>String</td><td>UID of actual Target</td></tr>
<tr><td>RefGuidTargetNew</td><td>String</td><td>New UID of destination Target</td></tr>
<tr><td>RefGuidSetDestination</td><td>String</td><td>UID of destination Set</td></tr>
<tr><td>IsShot</td><td>boolean</td><td>True if you want to copy Shot (Slot) configuration with the Target. False to copy only the target configuration.</td></tr>
<tr><td>IsCut</td><td>boolean</td><td>True if you want to move the target, false if you want to copy the target</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3">
<table>
<tr><td>ret</td><td>String</td><td>“DONE” if ok otherwise is an error</td></tr>
<tr><td>IsChangedProfile</td><td>boolean</td><td>True if the Target has changed the Profile</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetMoveCopyTarget", "params": {"RefGuidTarget":"632200ce-2145-4295-9236-0c459b3ac196","RefGuidTargetNew":"a8acf40f-48d2-4465-ac06-268b53d38b06","RefGuidSetDestination":"5482d20e-2304-41d1-8d2b-32adc2c314bc","IsShot":false,"IsCut":false,"UID":"39de4bd3-3fdf-4311-bf0b-fc0f24c450ab","MAC":"tDlsuiBw3sOWnJKerPMqDsK/jUw="}, "id": 42}

➔{"jsonrpc": "2.0", "result": 0, "id": 42}

➔

{"Event":"RemoteActionResult","Timestamp":1653133229.13705,"Host":"ORIONE","Inst":1,"UID":"39d

e4bd3-3fdf-4311-bf0b-fc0f24c450ab","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE", "IsChangedProfile":false}}

ff) RemoteRoboTargetMoveSet

Method	`RemoteRoboTargetMoveSet`		
Description	Move Set to another profile. Attention the target will become orphan and you must to fix this with the command itself if possible or manually with robotarget manager		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidSet	String	UID of Set
	DestinationProfile	String	File name with extension of the destination profile
	IsSequenceBlank	boolean	True if you want to leave the Base Sequence of the moved Target blank. You will need to fix this or you will have orphan targets
	IsSequenceDefault	boolean	True if you want to put the default sequence of the destination profile like new base sequence for the moved target
	IsSequenceFixed	boolean	True if you will specify for each target the match of base Sequence to use in DictTargetSequenza array
	DictTargetSequenza	Array of objects	List of target->base sequence changing to apply to the target in the new profile destination. One for each target
	TargetUID	string	UID of the Target
	SeqUID	string	UID of the Base Sequence to use for replace
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		

Remote Action Result Parameters in ParamRet Object	ret	String	“DONE” if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetMoveSet", "params": {"RefGuidSet":"7b28b0b5-4e69-436b-b363-cc3c99c7b069","DestionationProfile":"ColorTestAdvanced.v2y","IsSequenceBlank":false,"IsSequenceDefault":false,"IsSequenceFixed":false,"IsSequenceSelect":true,"SequenceOverrideUID":"","DictTargetSequenza":[{"TargetUID":"f32e23ed-2666-4c61-9b8f-f363bed455d2","SeqUID":"55b76f37-ad12-4a93-981a-03253ba46e22"}],"UID":"d95150fc-ce37-4b65-9952-3c3ae87fe727","MAC":"PJjmTjQF+uKXqTbgFmRc5gh3N48="}, "id": 63}

➔{"jsonrpc": "2.0", "result": 0, "id": 63}

➔
{"Event":"RemoteActionResult","Timestamp":1653134130.36958,"Host":"ORIONE","Inst":1,"UID":"d95150fc-ce37-4b65-9952-3c3ae87fe727","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

gg) RemoteRoboTargetCopyShot

Method	`RemoteRoboTargetCopyShot`		
Description	Copy Shot (Slot) configuration in another Target		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidShot	String	UID of Shot
	RefGuidShotNew	String	UID of the New Shot
	RefGuidTargetDestination	string	UID of the Target destination
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		

Remote Action Result Parameters in ParamRet Object	

ret	String	“DONE” if ok otherwise is an error
IsChangedProfile	boolean	True if the Slot has changed the Profile

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetCopyShot", "params": {"RefGuidShot":"14dca058-f03e-46e1-8a00-1d9f55c061c4","RefGuidShotNew":"fba54bc8-7bf2-4acd-9d19-14e29191e54f","RefGuidTargetDestination":"632200ce-2145-4295-9236-0c459b3ac196","UID":"07ba7155-0232-48c3-ba62-cf756a8bfd80","MAC":"yVqg9wo2oGhN08XCrx225iHi1bU="}, "id": 69}

➔{"jsonrpc": "2.0", "result": 0, "id": 69}

➔
{"Event":"RemoteActionResult","Timestamp":1653134435.8865,"Host":"ORIONE","Inst":1,"UID":"07ba7155-0232-48c3-ba62-cf756a8bfd80","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE","IsChangedProfile":false}}

hh) RemoteRoboTargetCopyTargetShot

Method	`RemoteRoboTargetCopyTargetShot`
Description	Copy all Shot (Slot) configuration of one Target to another Target
Params	

UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
RefGuidTarget	String	UID of Target
RefGuidTargetDestination	string	UID of the Target destination
MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*
Remote	

Action Result Parameters in ParamRet Object	ret	String	“DONE” if ok otherwise is an error
	IsChangedProfile	boolean	True if the Slot has changed the Profile

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetCopyTargetShot", "params": {"RefGuidTarget":"632200ce-2145-4295-9236-0c459b3ac196","RefGuidTargetDestination":"632200ce-2145-4295-9236-0c459b3ac196","UID":"b523e698-c91f-49e8-a503-616906ecccb0","MAC":"ByCXlQ0WUrMSepyzvoZZnfizFK8="}, "id": 80}

➔{"jsonrpc": "2.0", "result": 0, "id": 80}

➔
{"Event":"RemoteActionResult","Timestamp":1653135287.26637,"Host":"ORIONE","Inst":1,"UID":"b523e698-c91f-49e8-a503-616906ecccb0","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE","IsChangedProfile":false}}

ii) RemoteRoboTargetEnableDisableObject

Method	`RemoteRoboTargetEnableDisableObject`		
Description	Change enabled/disabled status to the object selected		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidObject	String	UID of Object
	ObjectType	integer	Object type: 0=Shot 1=Target 2=Set
	OperationType	itneger	Operation Type: 0=Enable 1=Disable
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in

			RemoteRoboTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		
Remote Action Result Parameters in ParamRet Object	ret	String	"DONE" if ok otherwise is an error
	IsChangedProfile	boolean	True if the Slot has changed the Profile

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetEnableDisableObject", "params": {"RefGuidObject":"14dca058-f03e-46e1-8a00-1d9f55c061c4","ObjectType":0,"OperationType":1,"UID":"1b6c9bff-1938-46e9-8456-9a58a05e4cfe","MAC":"d0pCB5p0SKJqRslfUPgM7CNRIuA="}, "id": 84}

➔{"jsonrpc": "2.0", "result": 0, "id": 84}

➔
{"Event":"RemoteActionResult","Timestamp":1653135850.36663,"Host":"ORIONE","Inst":1,"UID":"1b6c9bff-1938-46e9-8456-9a58a05e4cfe","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

jj) RemoteRoboTargetEnableDisableSetByTag

Method	RemoteRoboTargetEnableDisableSetByTag		
Description	Enable/Disable status of all the set having the selected tag (this command is cross profile)		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	Tag	String	Tag of Set
	OperationType	integer	Operation Type: 0=Enable 1=Disable
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.
Result	Integer(0)		

License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	

ret	String	“DONE” if ok otherwise is an error
IsChangedProfile	boolean	True if the Slot has changed the Profile

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetEnableDisableSetByTag", "params": {"Tag":"MyTag","OperationType":1,"UID":"1b6c9bff-1938-46e9-8456-9a58a05e4cfe","MAC":"d0pCB5p0SKJqRslfUPgM7CNRIuA="}, "id": 84}

➔{"jsonrpc": "2.0", "result": 0, "id": 84}

➔
{"Event":"RemoteActionResult","Timestamp":1653135850.36663,"Host":"ORIONE","Inst":1,"UID":"1b6c9bff-1938-46e9-8456-9a58a05e4cfe","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

kk) RemoteRoboTargetGetSessionListByTarget

Method	`RemoteRoboTargetGetSessionListByTarget`
Description	Return list of RoboTarget Session done for the Target
Params	

UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
RefGuidTarget	String	UID of Target.
MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.

Result	Integer(0)
License Required	*Advanced, Full*
Remote Action Result Parameters in ParamRet Object	

List	Array	Array of Session Objects

guid	String	UID of Session
datetimestart	datetime	Datetime of the Session start
datetimeend	datetime	Datetime of the Session end
repfilepdf	String	PDF report file if present

			refguidrun	string	UID of Run which session belong
			refguidtarget	string	UID of Target shot during Session
			result	integer	Session result, see table below
			status	integer	Session status 0=Idle 1=Running
			targetname	string	Name of the Target done during the session
			shotnumber	integer	Shot done for this session
			shotnumberdeleted		Shot done and delete for this session
			sessionexittext	string	Session exit string text

Session Result	Description
0 = UNDEF	Undefined
1 = OK	Session finished without error
2 = ABORTED	Session aborted
3 = FINISHED_ERROR	Session finished with error
4 = TIMEOUT	Session finished for timeout

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetSessionListByTarget", "params": {"RefGuidTarget":"632200ce-2145-4295-9236-0c459b3ac196","UID":"70b0f470-2023-44ff-b905-b7dc7d07b13d","MAC":"mfHowC5R4cTOze3RK5IxXAH6yB0="}, "id": 85}

←{"jsonrpc": "2.0", "result": 0, "id": 85}

←

{"Event":"RemoteActionResult","Timestamp":1653135947.45279,"Host":"ORIONE","Inst":1,"UID":"70b0f470-2023-44ff-b905-b7dc7d07b13d","ActionResultInt":4,"Motivo":"","ParamRet":{"list":[{ "guid":"4828e7bb-9f5d-4884-92cf-78b27d5d9c73", "datetimestart":1644630484, "datetimeend":1644631090, "repfilepdf":"", "refguidrun":"dcca3ed4-c1c6-4516-8290-2eca1aff412c", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":2, "status":0, "targetname":"Abell 28", "shotnumber":0, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"c02c4672-ea14-49d8-b6c7-1d408ec7bd6d", "datetimestart":1649555412, "datetimeend":1649563023, "repfilepdf":"", "refguidrun":"0fc349f5-6282-4f40-849d-6fa7ae6806b7", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":11, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"3dd5fe80-caa5-4f4d-ba19-fed2e50421e0", "datetimestart":1649625142, "datetimeend":1649626565, "repfilepdf":"",

"refguidrun":"8b863dc6-ebd1-4975-abc6-9193ed4d25db", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":2, "status":0, "targetname":"Abell 28", "shotnumber":1, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"dd75c481-2e4c-419c-a8f4-7c640ffcf775", "datetimestart":1649720232, "datetimeend":1649735343, "repfilepdf":"", "refguidrun":"1aaa9ad9-5619-4018-af6e-6a6c74db18fd", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":22, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"58ea8daf-5971-4390-a121-9375984bf0b5", "datetimestart":1649892552, "datetimeend":1649907663, "repfilepdf":"", "refguidrun":"716edfc7-5660-4366-b2a1-a0c9cebc0c69", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":22, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"139cebb4-0935-44ba-8a34-7ca62a9bae5a", "datetimestart":1649978711, "datetimeend":1649993823, "repfilepdf":"", "refguidrun":"fd1cb328-af95-45d8-bbef-16fb63f8fe1b", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":22, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"8db7df79-3c14-4d66-9960-e6a0dec37583", "datetimestart":1650161329, "datetimeend":1650162166, "repfilepdf":"", "refguidrun":"ce3d6eb9-3956-4d41-882d-43086d8acca6", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":2, "status":0, "targetname":"Abell 28", "shotnumber":0, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"9cda9c06-fa33-4426-925b-b1bef885c6b8", "datetimestart":1650237191, "datetimeend":1650252303, "repfilepdf":"", "refguidrun":"765873b7-65f8-41a2-8182-54a8edf1d26a", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":22, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"d2adfead-04f3-4d02-a2fc-8a0a23b0fd12", "datetimestart":1650324851, "datetimeend":1650338464, "repfilepdf":"", "refguidrun":"91d9548b-45eb-4887-b764-00cc77a8e809", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":20, "shotnumberdeleted":0, "sessionexittext":"" },{ "guid":"1023e537-11bf-4a56-a887-a02879cb995e", "datetimestart":1650415642, "datetimeend":1650415758, "repfilepdf":"", "refguidrun":"365f910e-99b1-42b5-aecc-c000cec32475", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":3, "status":0, "targetname":"Abell 28", "shotnumber":0, "shotnumberdeleted":0, "sessionexittext":"Sequence Finished with Error - Begin Precise Pointing Abell 28 (Plate Solving (With Blind) Actual Location Error (Blind Solving Error ()))" },{ "guid":"1b410c71-24c0-4155-a17f-dc337f5707a4", "datetimestart":1650416068, "datetimeend":1650424623, "repfilepdf":"", "refguidrun":"365f910e-99b1-42b5-aecc-c000cec32475", "refguidtarget":"632200ce-2145-4295-9236-0c459b3ac196", "result":1, "status":0, "targetname":"Abell 28", "shotnumber":10, "shotnumberdeleted":0, "sessionexittext":"" }]}}

ll) RemoteRoboTargetGetSessionContainerCountByTarget

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetSessionContainerCountByTarget</td></tr>
<tr><td>Description</td><td colspan="3">Return list of RoboTarget Session done for the Target</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidTarget</td><td>String</td><td>UID of Target.</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td>JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoboTargetGetSet.</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>Array</td><td>Array of Session Container
<table>
<tr><td>Duration</td><td>integer</td><td>Duration of All Session in seconds</td></tr>
<tr><td>Progress</td><td>datetime</td><td>Percentage of progress for the target. In case of Preset Time Interval Target with Progress Reset Flag this value rappresent the progress obtained in the last runned or running interval only.</td></tr>
<tr><td>SessionCount</td><td>datetime</td><td>Count of Sessione done for the Target</td></tr>
<tr><td>ShotDoneCount</td><td>String</td><td>Shot Done for the Target and not removed</td></tr>
<tr><td>ShotDoneRawCount</td><td>string</td><td>Shot Done for the target , included the removed</td></tr>
<tr><td>ShotRequested</td><td>string</td><td>Shot Requested for the target</td></tr>
</table></td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetSessionContainerCountByTarget", "params": {"RefGuidTarget":"632200ce-2145-4295-9236-0c459b3ac196","UID":"e921c04d-e588-4d03-a651-01db47764efb","MAC":"tBHbLc6x2drvjMdVcYRIIkZo450="}, "id": 82}

←{"jsonrpc": "2.0", "result": 0, "id": 82}

←

{"Event":"RemoteActionResult","Timestamp":1653135287.82817,"Host":"ORIONE","Inst":1,"UID":"e921c04d-e588-4d03-a651-01db47764efb","ActionResultInt":4,"Motivo":"","ParamRet":{"data":{"Duration":93207,"Progress":23,"SessionCount":11,"ShotDoneCount":130,"ShotDoneRawCount":130,"ShotRequested":576}}}

mm) RemoteRoboTargetGetRuns

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetRuns</td></tr>
<tr><td>Description</td><td colspan="3">Return list of Runs done for Profile (all and last 30days)</td></tr>
<tr><td>Params</td><td colspan="3">
<table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>ProfileName</td><td>String</td><td>Profile name used for search about Runs. If empty will be answered the Runs for all profile configured in Voyager</td></tr>
<tr><td>ListDays</td><td>array</td><td>Array of integers : 0 for all days, 30 for last 30 days etc etc</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table>
</td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td>List</td><td>Array</td><td>Array of Run Objects
<table>
<tr><td>guid</td><td>string</td><td>UID of Object</td></tr>
<tr><td>profilename</td><td>String</td><td>Named of profile name which Runs belong</td></tr>
<tr><td>count</td><td>array</td><td>Array of counters</td></tr>
<tr><td>days</td><td>integer</td><td>Number of last days to now (0 days means all runs)</td></tr>
<tr><td>runs</td><td>integer</td><td>Number of runs</td></tr>
</table>
</td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetRuns", "params": {"ProfileName":"","ListDays":[],"UID":"a97bd5b6-ee7c-40f6-8601-0ee743df040e","MAC":"/sBDfoeaiD+jdmirMClEm4i/nJQ="}, "id": 8}

➔{"jsonrpc": "2.0", "result": 0, "id": 8}

➔
{"Event":"RemoteActionResult","Timestamp":1653120334.38001,"Host":"ORIONE","Inst":1,"UID":"a97bd5b6-ee7c-40f6-8601-0ee743df040e","ActionResultInt":4,"Motivo":"","ParamRet":{"list":[{"guid":"18b00ae5-

5b92-4116-8bd1-70b7df07cc3d", "profilename":"ColorTestAdvanced.v2y", "count":[{"days":0,"runs":0},{"days":30,"runs":0}]},{"guid":"92c74339-c6a7-4a25-89b2-1ebf9c80480b", "profilename":"temp.v2y", "count":[{"days":0,"runs":0},{"days":30,"runs":0}]},{"guid":"8def0709-c508-47ea-91c3-0d9fd6ebe98b", "profilename":"TestFlatNoMount.v2y", "count":[{"days":0,"runs":25},{"days":30,"runs":25}]}]}}

nn) RemoteRoboTargetGetShotJpg

<table>
<tr><td>Method</td><td colspan="3">RemoteRoboTargetGetShotJpg</td></tr>
<tr><td>Description</td><td colspan="3">Return the base64 Jpeg image if in cache or available on disk</td></tr>
<tr><td>Params</td><td colspan="3"><table>
<tr><td>UID</td><td>String</td><td>Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated</td></tr>
<tr><td>RefGuidShotDone</td><td>String</td><td>UID of Shot Done, empty if you want to search by file name</td></tr>
<tr><td>FITFileName</td><td>String</td><td>Only FIT File Name with extension (no path), empty if you want to search by UID</td></tr>
<tr><td>MAC</td><td>String</td><td>Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.</td></tr>
</table></td></tr>
<tr><td>Result</td><td colspan="3">Integer(0)</td></tr>
<tr><td>License Required</td><td colspan="3">Advanced, Full</td></tr>
<tr><td>Remote Action Result Parameters in ParamRet Object</td><td colspan="3"><table>
<tr><td>ret</td><td>String</td><td>“DONE” if ok otherwise is an error</td></tr>
<tr><td>HFD</td><td>boolean</td><td>True if the Slot has changed the Profile</td></tr>
<tr><td>StarIndex</td><td>numeric</td><td>Star Index of image</td></tr>
<tr><td>PixelDimX</td><td>integer</td><td>Dimension on x axis in pixels</td></tr>
<tr><td>PixelDimY</td><td>integer</td><td>Dimension on y axis in pixels</td></tr>
<tr><td>Min</td><td>numeric</td><td>Min ADU value of image</td></tr>
<tr><td>Max</td><td>numeric</td><td>Max ADU value of image</td></tr>
<tr><td>Mean</td><td>numeric</td><td>Mean ADU value of image</td></tr>
<tr><td>BaseData</td><td>string</td><td>Base64 data of JPG file image</td></tr>
</table></td></tr>
</table>

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetGetShotJpg", "params": {"RefGuidShotDone":"ddeea856-8ba7-41be-9689-a20445c022d4","FITFileName":"","UID":"a97bd5b6-ee7c-40f6-8601-0ee743df040e","MAC":"/sBDfoeaiD+jdmirMClEm4i/nJQ="}, "id": 8}

➔{"jsonrpc": "2.0", "result": 0, "id": 8}

🡐

{"Event":"RemoteActionResult","Timestamp":1653234141.8514,"Host":"ORIONE","Inst":1,"UID":"a97bd5b6-ee7c-40f6-8601-0ee743df040e","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE","HFD":6.67885140282118,"StarIndex":6.67885140282118,"PixelDimX":9576,"PixelDimY":6388,"Min":573,"Max":65535,"Mean":1236,"Base64Data":"/9j/4AAQSkZJRgABAQEAYABgAAD/2wBDABwTFRgVERwYFhgfHRwhKUUtKSYmKVQ8QDJFZFhpZ2JYYF9ufJ6GbnWWd19giruLlqOpsbOxa4TC0MGszp6usar/2wBDAR0fHykkKVEtLVGqcmByqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqqr/wAARCArvEGQDASIAAhEBAxEB/8QAHwAAAQUBAQEBAQEAAAAAAAAAAAECAwQFBgcICQoL/8QAtRAAAgEDAwIEAwUFBAQAAAF9AQIDAAQRBRIhMUEGE1FhByJxFDKBkaEII0KxwRVS0fAkM2JyggkKFhcYGRolJicoKSo0NTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqDhIWGh4iJipKTlJWWl5iZmqKjpKWmp6ipqrKztLW2t7i5usLDxMXGx8jJytLT1NXW19jZ2uHi4+Tl5ufo6erx8vP09fb3+Pn6/8QAHwEAAwEBAQEBAQEBAQAAAAAAAAECAwQFBgcICQoL/8QAtREAAgECBAQDBAcFBAQAAQJ3AAECAxEEBSExBhJBUQdhcRMiMoEIFEKRobHBCSMzUvAVYnLRChYkNOEl8RcYGRomJygpKjU2Nzg5OkNERUZHSElKU1RVVldYWVpjZGVmZ2hpanN0dXZ3e ......

oo) RemoteRoboTargetAbort

Method	RemoteRoboTargetAbort		
Description	Abort Remote RoboTarget Seuence based in Target UID or Set UID or Target TAG or Set TAG. If all the parameters are empty (equal to “”) the RoboTarget entire Action will be aborted		
Params	UID	String	Unique identifier of the Action to abort. Use a Guide Window identifier or a unique key string generated
	RefGuidTarget	String	UID of Target (if target running in remote action matches the sequence will be aborted) or empty
	RegGuidSet	String	UID of Set (if target running in remote action matches one of the targets in the Set the sequence will be aborted) or empty
	RegGuidTargetTag	String	TAG of Target (if target running in remote action matches the sequence will be aborted) or empty
	RegGuidSetTag	String	TAG of Set (if target running in remote action matches one of the targets in the Set the sequence will be aborted) or empty
	MAC	String	Create a concatenated string with RoboTarget Shared secret + SessionKey (the Timestamp string received in the Event Version sent by the Server as is)) + ID of JSON-RPC command + UID of Voyager Command. Finally make an SHA1 hash and convert to base 64 string, see the example in RemoteRoooTargetGetSet.
Result	Integer(0)		
License Required	*Advanced, Full*		

Remote Action Result Parameters in ParamRet Object	ret	String	“DONE” if ok otherwise is an error

(*) hash reported in the example are only for didattical scope and the final MAC are not correct

➔ {"method": "RemoteRoboTargetAbort", "params": {"RefGuidTarget":"","RefGuidSet":"","RefGuidTargetTag":"","RefGuidSetTag":"","UID":"862fad7a-eb74-43f5-85f6-d8eb4331675e","MAC":"c6xXH2Ou6IIa4oeirnns5ObEA4oY="}, "id": 14}

🡐{"jsonrpc": "2.0", "result": 0, "id": 14}

🡐
{"Event":"RemoteActionResult","Timestamp":1671453897.24351,"Host":"ORIONE","Inst":1,"UID":"862fad7a-eb74-43f5-85f6-d8eb4331675e","ActionResultInt":4,"Motivo":"","ParamRet":{"ret":"DONE"}}

7. Preset Time Interval JSON Object

This is the JSON array object for definition and retrieve of the Preset Time Intervals

UID	string	UID of the Interval Object
Enabled	boolean	Define if the interval is enabled or not. Disabled means not used for scheduling
DateTimeStart	time	Epoch 1970 time expressed for the start of interval
DateTimeEnd	time	Epoch 1970 time expressed for the end of interval

Status	integer	Status of the Interval
		`IDLE = 0`
		`FINISHED = 1`
		`EXPIRED = 2`

Interval greater than 12Hr will not be used, interval out of the civil/nautical/astronomical night will be truncated.

Example:

[{"UID":"1CAA031D-D0B1-498E-8C5C-75E7EA6A29FC","Enabled":true,"DateTimeStart":1670081444,"DateTimeEnd":1670092244,"Status":0},{"UID":"C1BBAD2F-2314-46CE-8FAE-594CC1000EBA","Enabled":false,"DateTimeStart":1670094703,"DateTimeEnd":1670106771,"Status":0}]

8. Open RoboTarget API

A series of RoboTarget APIs are available in the normal Application Server and are disclosed to all, this is the list:

- `RemoteOpenRoboTargetGetTargetList`
- `RemoteOpenRoboTargetGetShotDoneList`
- `RemoteOpenRoboTargetSetShotDoneRating`
- `RemoteOpenRoboTargetRemoveShotDone`
- `RemoteOpenRoboTargetRestoreShotDone`
- `RemoteOpenRoboTargetUpdateBulkShotDone`
- `RemoteOpenRoboTargetSetShotDoneRatingByFileName`
- `RemoteOpenRoboTargetRemoveShotDoneByFileName`
- `RemoteOpenRoboTargetRestoreShotDoneByFileName`

More info about in the Application Server documentation in the dedicated paragraph.