

QIAcuity Software Suite API 3.2.0.0 (build 110.0.0.335)

Contents

1	Usage	7
2	Identifying instruments, drawers and plate slots in the API	9
3	Channels in imaging	9
4	Commands	10
4.1	BookDrawer	10
4.2	ReleaseDrawerBooking	10
4.3	OpenDrawer	10
4.4	CloseDrawer	10
4.5	RunExperiment	10
5	Events	10
5.1	DRAWER_BOOKED	11
5.2	DRAWER_OPENED	11
5.3	DRAWER_NOT_OPENED	12
5.4	DRAWER_OPENED_MANUALLY	12
5.5	DRAWER_CLOSED	13
5.6	DRAWER_NOT_CLOSED	13
5.7	DRAWER_CLOSED_MANUALLY	14
5.8	DRAWER_BOOKING_RELEASED	14
5.9	DRAWER_BOOKING_NOT_RELEASED	14
5.10	EXPERIMENT_PROCESSING_STARTED	14
5.11	EXPERIMENT_READY	15
5.12	EXPERIMENT_PROGRESS	15
5.13	EXPERIMENT_ABORTED	16

6	Sequence to run an experiment	18
7	Authentication	19
8	Monitoring	19
9	Setting thresholds via analysis templates	19
9.1	Overview	19
9.1.1	Step 1 - overriding the thresholds initially set by the Software Suite	19
9.1.2	Step 2 - validation of the quality of the results	20
9.1.3	Step 3 - setting up dynamic thresholds	20
9.2	Identifying rules in the experiment results	21
9.3	Manual changes to the thresholds	21
9.4	Automatic reanalysis	21
10	Changelog	21
10.1	3.1.1.0	21
10.2	3.2.0.0	22
11	Running an experiment – an end-to-end scenario example with real API calls	22
11.1	Define an experiment	22
11.1.1	Option A – define an experiment based on an existing template	22
11.1.2	Option B – define an experiment from scratch	23
11.2	Put the plate into the instrument	25
11.3	Run the experiment	27
11.4	[Optionally] Release the booking	28
11.5	Wait for the results to be ready	28
11.6	Download the results	28
12	Endpoints	29
12.1	AddExperimentParameters	29
12.1.1	Request body example	29
12.1.2	Request body properties	31
12.1.3	Example responses	31
12.1.4	Responses	31
12.2	AddDpcrSteps	31
12.2.1	Request body example	32
12.2.2	Request body properties	33
12.2.3	Example responses	33
12.2.4	Responses	33

12.3	GetRfuDetails	34
12.3.1	Request body example	34
12.3.2	Requet body properties	34
12.3.3	Example responses	35
12.3.4	Responses	35
12.4	AssignAnalysisTemplate	35
12.4.1	Request body example	35
12.4.2	Requet body properties	37
12.4.3	Example responses	37
12.4.4	Responses	38
12.5	DefineExperiment	38
12.5.1	Request body example	39
12.5.2	Requet body properties	40
12.5.3	Example responses	40
12.5.4	Responses	40
12.6	DefineExperimentFromTemplate	40
12.6.1	Request body example	41
12.6.2	Requet body properties	41
12.6.3	Example responses	41
12.6.4	Responses	41
12.7	RunExperiment	42
12.7.1	Request body example	42
12.7.2	Requet body properties	42
12.7.3	Example responses	42
12.7.4	Responses	42
12.8	ReleaseDrawerBooking	43
12.8.1	Request body example	43
12.8.2	Requet body properties	43
12.8.3	Example responses	43
12.8.4	Responses	43
12.9	OpenDrawer	44
12.9.1	Request body example	44
12.9.2	Requet body properties	44
12.9.3	Example responses	44
12.9.4	Responses	44
12.10	CloseDrawer	45
12.10.1	Request body example	45
12.10.2	Requet body properties	45
12.10.3	Example responses	45

12.10.4 Responses	46
12.11 BookDrawer	46
12.11.1 Request body example	46
12.11.2 Requeset body properties	46
12.11.3 Example responses	46
12.11.4 Responses	47
12.12 GetInstruments	47
12.12.1 Example responses	47
12.12.2 Responses	48
12.12.3 Response Schema	48
12.13 HealthCheck	49
12.13.1 Example responses	49
12.13.2 Responses	50
12.13.3 Response Schema	50
12.14 GetExperimentStatus	50
12.14.1 Requeset body properties	50
12.14.2 Example responses	51
12.14.3 Responses	51
12.15 GetExperimentResults	51
12.15.1 Requeset body properties	51
12.15.2 Example responses	52
12.15.3 Responses	54
12.15.4 Response Schema	55
12.16 GetPrimingProfiles	62
12.16.1 Requeset body properties	63
12.16.2 Example responses	63
12.16.3 Responses	63
12.16.4 Response Schema	63
12.17 GetPlateTypes	64
12.17.1 Example responses	64
12.17.2 Responses	64
12.17.3 Response Schema	64
12.18 GetDyes	65
12.18.1 Example responses	65
12.18.2 Responses	65
12.18.3 Response Schema	65
12.19 GetEvent	66
12.19.1 Example responses	66
12.19.2 Responses	66

12.20	AcknowledgeEvent	66
12.20.1	Request body properties	67
12.20.2	Example responses	67
12.20.3	Responses	68
12.21	DeleteExperiment	68
12.21.1	Request body properties	68
12.21.2	Example responses	68
12.21.3	Responses	68

13 Endpoint schemas 69

13.1	AddParametersToExperimentRequest	69
13.2	AddParametersToExperimentRequest.Reaction	71
13.3	AddParametersToExperimentRequest.Reaction.Target	72
13.4	AddParametersToExperimentRequest.SampleControl	72
13.5	AddParametersToExperimentRequest.SampleControl.Control	73
13.6	AddParametersToExperimentRequest.SampleControl.NonTemplateControl	74
13.7	AddParametersToExperimentRequest.SampleControl.Sample	74
13.8	AddParametersToExperimentRequest.Well	75
13.9	AddParametersToExperimentRequest.Well.ControlType	77
13.10	AddParametersToExperimentRequest.Well.Sample	78
13.11	ChannelSliceDefinition	79
13.12	ErrorResponse	79
13.13	ValidationErrorResponse	80
13.14	AddDPCRStepsToExperimentRequest	81
13.15	AddDPCRStepsToExperimentRequest.Cycling	82
13.16	AddDPCRStepsToExperimentRequest.Cycling.Cycle	83
13.17	AddDPCRStepsToExperimentRequest.Cycling.Cycle.CycleStep	83
13.18	AddDPCRStepsToExperimentRequest.Imaging	84
13.19	AddDPCRStepsToExperimentRequest.Imaging.ImagingProfile	85
13.20	ChannelDefinition	85
13.21	RfuRequest	86
13.22	StreamingResponseBody	87
13.23	AssignAnalysisTemplateRequest	87
13.24	AssignAnalysisTemplateRequest.AmplitudeMultiplexing	89
13.25	AssignAnalysisTemplateRequest.AmplitudeMultiplexingAutoThreshold	90
13.26	AssignAnalysisTemplateRequest.DynamicThresholdRule	90
13.27	AssignAnalysisTemplateRequest.ForNegativeControl	92
13.28	AssignAnalysisTemplateRequest.ForNonTemplateControl	92
13.29	AssignAnalysisTemplateRequest.ForPositiveControl	92

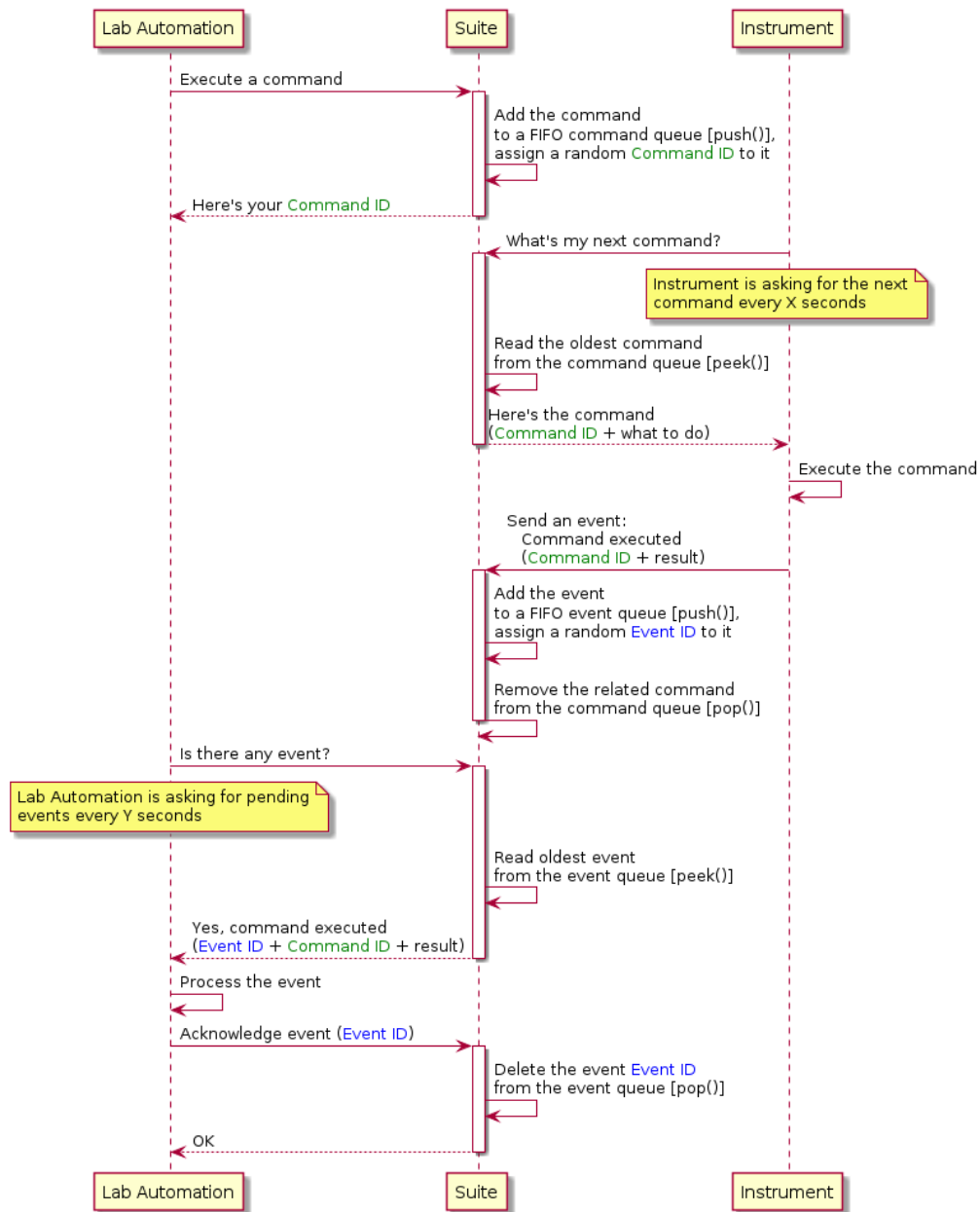
13.30AssignAnalysisTemplateRequest.InitialAutoThresholdOverrideRule	93
13.31AssignAnalysisTemplateRequest.ReactionMixAnalysisRules	94
13.32AssignAnalysisTemplateRequest.SingleThreshold	95
13.33AssignAnalysisTemplateRequest.Threshold	96
13.34AssignAnalysisTemplateRequest.ValidationRules	96
13.35Rule	96
13.36DefineExperimentRequest	97
13.37DefineExperimentRequest.DPCRRunSteps	98
13.38DefineExperimentRequest.DPCRRunSteps.Cycling	100
13.39DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle	101
13.40DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle.CycleStep	101
13.41DefineExperimentRequest.DPCRRunSteps.Imaging	102
13.42DefineExperimentRequest.DPCRRunSteps.Imaging.ImagingProfile	102
13.43DefineExperimentRequest.DPCRRunSteps.Priming	103
13.44DefineExperimentFromTemplateRequest	103
13.45RunExperimentRequest	104
13.46ReleaseDrawerBookingRequest	105
13.47OpenDrawerRequest	105
13.48CloseDrawerRequest	106
13.49BookDrawerRequest	106
13.50InstrumentStateResponse	106
13.51InstrumentStateResponse.DrawerState	107
13.52HealthCheck	108
13.53ExperimentStatusResponse	108
13.54AbsoluteQuantificationResult	109
13.55AbsoluteQuantificationResult.ConcentrationPerChannelWithMean	112
13.56AbsoluteQuantificationResult.ConcentrationPerChannelWithMean.Target	117
13.57AbsoluteQuantificationResult.Reaction	117
13.58AbsoluteQuantificationResult.Sample	118
13.59AbsoluteQuantificationResult.Well	119
13.60ConcentrationWithErrorCode	119
13.61DispersionMeasureResponse	121
13.62ExperimentResultsResponse	121
13.63MinimumValidPartitions	124
13.64QualityAssuranceValidationRuleResults	125
13.65SignalThreshold	125
13.66SignalThreshold.AmplitudeMultiplexing	125
13.67SignalThreshold.SingleThreshold	126
13.68ThresholdSource	127

13.69ThresholdSource.Auto	127
13.70ThresholdSource.AutomationRule	127
13.71ThresholdSource.Lasso	128
13.72ThresholdSource.Manual	128
13.73EventResponse	128
13.74JsonNode	129

1 Usage

When controlling the instruments, the Lab Automation Service uses a concept of command and event queues. Both queues are working in a “first in, first out” fashion which means the oldest element in the queue will be retrieved (processed) first.

Commands tell the QIAcuity instrument what to do. Events tell what the instrument or Software Suite has done. The Lab Automation API provides endpoints for adding and retrieving commands and events.



2 Identifying instruments, drawers and plate slots in the API

There are three models of the QIAcuity instrument: QIAcuity One, QIAcuity Four and QIAcuity Eight.

The models differ by the number of drawers and the number of slots for plates in the drawers.

Whenever the API requires to provide the model of the instrument or identifiers of drawers and slots, the following values must be used:

Note: slots are counted from the left when standing in front of the instrument

Instrument model	Model type code	Identifiers of drawers and slots
QIAcuity One	P1	- Drawer0 , slot 0
QIAcuity Four	P4	- Drawer0 , slots 0, 1, 2, 3
QIAcuity Eight	P8	- upper drawer: Drawer0 , slots 0, 1, 2, 3 - lower drawer: Drawer1 , slots 0, 1, 2, 3

3 Channels in imaging

Imaging channels are defined by the colors of the excitation and emission filters. Some of the filter combinations require the High-Multiplexing-Reference Channel to be enabled in the imaging step.

The following filter combinations are supported:

Excitation	Emission	High Multiplexing Reference Channel required?
GREEN	GREEN	No
YELLOW	YELLOW	No
ORANGE	ORANGE	No
RED	RED	No
CRIMSON	CRIMSON	No
FARRED	FARRED	Yes
GREEN	YELLOW	Yes
YELLOW	ORANGE	Yes
ORANGE	RED	Yes
RED	CRIMSON	Yes
CRIMSON	FARRED	Yes

4 Commands

Each of the following commands has a corresponding API endpoint with the same name. The parameters are specified in the description of the particular endpoint.

4.1 BookDrawer

Books particular drawers for exclusive robot use.

In order to prevent a human to open the drawer while the Lab Automation commands the robot to insert/remove plate(s) to/from the instrument, the Lab Automation must book a drawer for exclusive use. Afterwards, manual operation of the drawer via a physical button or a button on the Instrument's UI is disabled. In order to enable it again, ReleaseDrawerBooking endpoint must be called.

4.2 ReleaseDrawerBooking

Releases the exclusive booking of the drawers specified in the command.

Afterwards, manual operation of the drawer via a physical button or a button on the Instrument's UI is enabled.

Tip: in fully-automated use scenarios, there is no need to release the booking at all. Releasing the booking is only required if a human needs to interact with the drawer.

4.3 OpenDrawer

Opens the drawer specified in the command. The drawer must be booked beforehand.

4.4 CloseDrawer

Closes the drawer specified in the command. The drawer must be booked beforehand.

4.5 RunExperiment

Links a physical plate to an experiment definition in Software Suite, then starts the experiment. The drawer must be booked beforehand.

5 Events

Events can be received in the `/lab-automation/v1/events` endpoint.

Events are available in two forms:

- command-related - results of executing a command;
- unsolicited - side-effects of actions happening in the background of Software Suite.

Events may contain payloads. To make it possible to change the schema of the payloads and recognize changed schema by the consumers of the API, the payloads are versioned by a `payloadSchemaVersion` property.

Example of a response from the `/lab-automation/v1/events` endpoint.

```
{
  "id": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "commandId": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "instrumentId": "instrument1234",
  "type": "DRAWER_BOOKED",
  "payloadSchemaVersion": 1,
  "payload": {
    "freeSlotsInDrawers": {
      "Drawer0": [0,1,2,3],
      "Drawer1": [0,1,2,3]
    }
  }
}
```

In case of unsolicited events, the `commandId` property is `null`.

5.1 DRAWER_BOOKED

Event for the `BookDrawer` command:

- When drawer was successfully booked
- When drawer was already booked and Lab Automation asked for it once again

Payload example:

```
{
  "freeSlotsInDrawers": {
    "Drawer0": [0, 1, 2, 3],
    "Drawer1": [0, 1, 2, 3]
  }
}
```

Schema version: 1

5.2 DRAWER_OPENED

Event for the `OpenDrawer` command:

- When booking for the requested drawer is active and the drawer was successfully opened
- When booking for the requested drawer is active and the drawer was already open

Payload example:

```
{
  "drawerName": "Drawer1",
  "freeSlotsInDrawers": {
    "Drawer0": [0, 1, 2, 3],
    "Drawer1": [0, 1, 2, 3]
  }
}
```

Schema version: 1

5.3 DRAWER_NOT_OPENED

Event for the `OpenDrawer` command:

- When Lab Automation Software does not have an active booking for the requested drawer

Payload example:

```
{
  "drawerName": "Drawer1",
  "reason": "NO_ACTIVE_BOOKING"
}
```

`reason` is one of:

- UNKNOWN_ISSUE
- INVALID_MODULE_ID (drawer with specified name does not exist)
- NO_ACTIVE_BOOKING (drawer is not booked by Lab Automation Software)
- OTHER_DRAWER_OPENED_BY_COMMAND (for instruments with more than one drawer: if another drawer is already open)

Schema version: 1

5.4 DRAWER_OPENED_MANUALLY

Unsolicited event occurring when a human opened a drawer manually.

Payload example:

```
{  
  "drawerName": "Drawer1"  
}
```

Schema version: 1

5.5 DRAWER_CLOSED

Event for the `CloseDrawer` endpoint:

- When booking for the requested drawer is active, the drawer was successfully closed and plates on the drawer were identified
- When booking for the requested drawer is active and the drawer was already closed

Payload example:

```
{  
  "drawerName": "Drawer1"  
}
```

Schema version: 1

5.6 DRAWER_NOT_CLOSED

Event for the `CloseDrawer` endpoint:

- When Lab Automation Software does not have an active booking for the requested drawer

Payload example:

```
{  
  "drawerName": "Drawer1",  
  "reason": "NO_ACTIVE_BOOKING"  
}
```

`reason` is one of:

- UNKNOWN_ISSUE
- INVALID_MODULE_ID (drawer with specified name does not exist)
- NO_ACTIVE_BOOKING (drawer is not booked by Lab Automation Software)
- OTHER_DRAWER_OPENED_BY_COMMAND (for instruments with more than one drawer: if another drawer is already open)

Schema version: 1

5.7 DRAWER_CLOSED_MANUALLY

Unsolicited event occurring when a human closed a drawer manually.

Payload example:

```
{  
  "drawerName": "Drawer1"  
}
```

Schema version: 1

5.8 DRAWER_BOOKING_RELEASED

Event for the `ReleaseDrawerBooking` command when any of the following happens:

- booking was released
- Lab Automation Software asked for releasing more bookings than it owned
- there was nothing to release (no bookings were active)

Payload example:

```
{  
  "drawerName": "Drawer1"  
}
```

Schema version: 1

5.9 DRAWER_BOOKING_NOT_RELEASED

Event for the `ReleaseDrawerBooking` command when Lab Automation owns a booking for the requested drawer and the drawer is open

Payload is `null`.

Schema version: 1

5.10 EXPERIMENT_PROCESSING_STARTED

Event for the `RunExperiment` command when instrument started the experiment.

Payload is `null`.

Schema version: 1

5.11 EXPERIMENT_READY

Unsolicited event created when for all configured imaging steps in the experiment:

- all wells are analyzed by Suite (including analysis template) OR
- instrument notified that image transfer is finished for a particular imaging step and all the wells for which pictures were received were analyzed by Suite (including analysis template)

Payload:

```
{
  "plateId": "7da11fca-31e9-4bdd-9ed4-2a47eacefcf0",
  "imagingStepIndexes": [3],
  "allImagingStepIndexes": [3, 4],
  "allImagingStepsReady": false
}
```

The `allImagingStepsReady` flag will be `true` if `imagingStepIndexes` contains all elements from `allImagingStepIndexes`.

Note: this event may be sent multiple times. In a successful run scenario it will be sent twice:

- when images are received for all wells and the analysis is finished;
- when the instrument confirms that the image transfer is finished for all wells and the analysis is finished.

It will also be sent in case of reanalysis - see Automatic reanalysis section in the Analysis templates section.

In a fully automated scenario, it is enough to wait for the first event with `allImagingStepsReady` flag set to `true`. Subsequent `EXPERIMENT_READY` events can be then safely ignored.

Schema version: 3

5.12 EXPERIMENT_PROGRESS

Unsolicited event created when the instrument notifies Suite about the current step of the experiment.

Payload example:

```
{
  "plateId": "7da11fca-31e9-4bdd-9ed4-2a47eacefcf0",
  "runStepIndex": 2,
  "experimentStatus": "IMAGING_STARTED"
}
```

`experimentStatus` is one of:

- `RUN_STARTED`

- PRIMING_STARTED
- PRIMING_COMPLETED
- CYCLING_STARTED
- CYCLING_COMPLETED
- IMAGING_STARTED
- IMAGE_TRANSFER_STARTED
- IMAGE_TRANSFER_COMPLETED
- IMAGING_COMPLETED
- RUN_FAILED
- RUN_STOPPED
- RUN_COMPLETED

Schema version: 1

5.13 EXPERIMENT_ABORTED

Event for the `RunExperiment` command when the instrument cannot start the experiment due to the reason specified in the Payload.

Payload example:

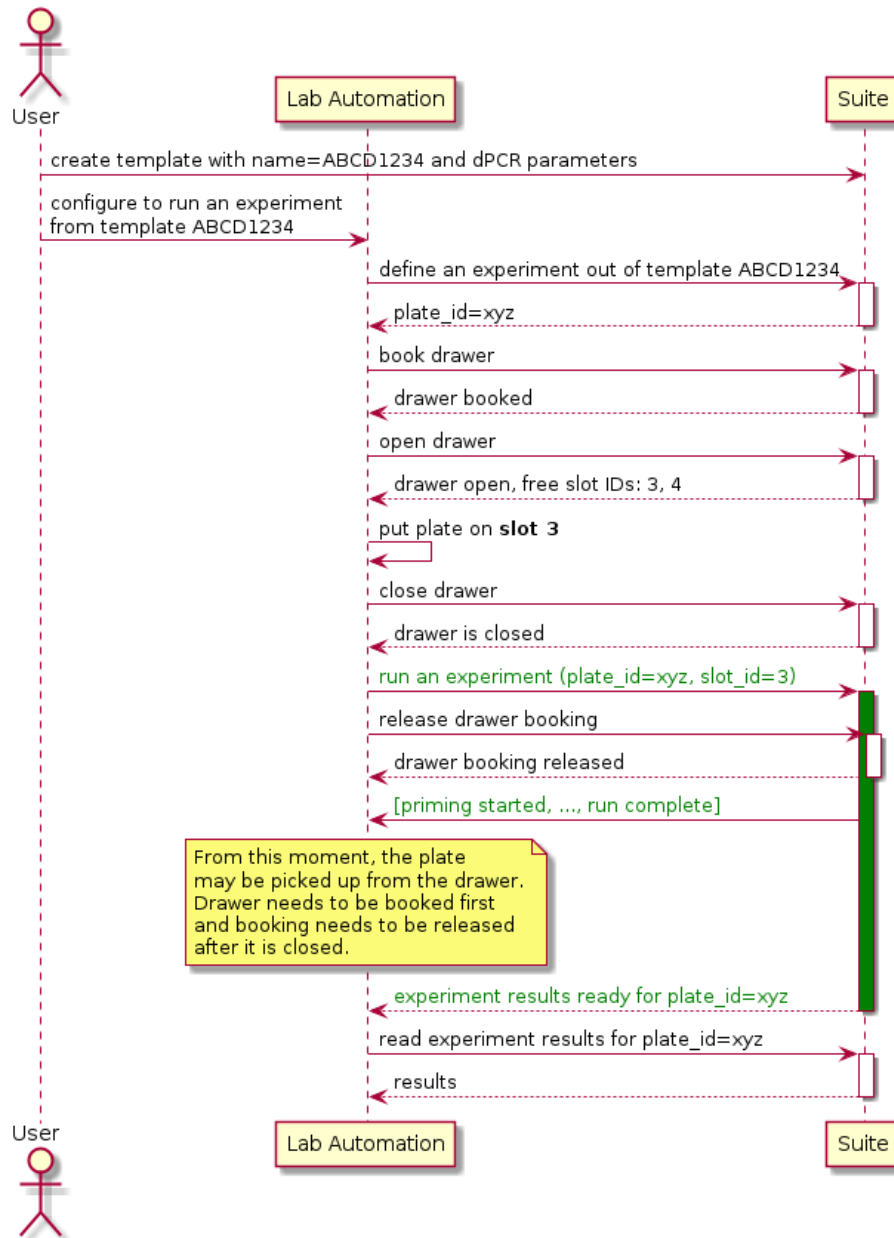
```
{
  "reason": "PLATE_INVALID_STATE"
}
```

`reason` is one of:

- ISSUE_WITH_LINKING_PLATE (could not link a plate from the drawer to a plate in Software Suite, contact QIAGEN support)
- UNKNOWN_ISSUE
- INVALID_MODULE_ID (drawer with specified name does not exist)
- NO_ACTIVE_BOOKING (drawer is not booked by Lab Automation Software)
- NO_PLATE (the plate does not have a barcode)
- PLATE_INVALID_STATE (attempting to run an experiment on a plate which has already been aborted during priming)
- NO_MATCHING_BARCODES (the barcode of the provided plate does not match the barcode specified for this plate in Software Suite)
- NO_ENOUGH_DISK_SPACE (there is not enough disk space on the instrument to run the experiment)

Schema version: 1

6 Sequence to run an experiment



7 Authentication

The API is secured by an API key. The key can be obtained in QIAcuity Software Suite. Please refer to the User Manual for instructions.

The key must be included in every request to the API in the **Authorization** header with an **ApiKey** prefix. Example: **Authorization: ApiKey <your_api_key>**.

This authentication method is later referred to as *Lab Automation API Key Authorization*.

8 Monitoring

Lab Automation API provides two endpoints to monitor the state of the system:

- **GetInstruments** to enumerate the instruments registered in Suite, see their online status and state of their drawers
- **HealthCheck** to see how many events are in the command and event queues per instrument

9 Setting thresholds via analysis templates

9.1 Overview

Analysis templates allow to automate setting thresholds via the **AssignAnalysisTemplate** endpoint. The automation happens in three steps:

1. Overriding the thresholds initially set by the Software Suite;
2. Validation of the quality of the results;
3. Setting up dynamic thresholds based on thresholds from controls or NTCs.

Each of those steps is optional.

The automation steps are configured per reaction mix as a set of rules. In the API request, reaction mixes are identified by their names.

When the rules are applied to the experiment results, **all wells** from **all imaging steps** are processed, even if they are not part of a reaction mix specified in the analysis template. As a consequence, **any manual changes to the thresholds in any of the wells will be overwritten by step 1**.

9.1.1 Step 1 - overriding the thresholds initially set by the Software Suite

This step allows to set fixed thresholds per channel. At the same time, it allows to set the threshold mode to **ST** (Single Threshold) or **AM** (Amplitude Multiplexing). It is also possible to set it to use the automatic threshold for **AM** mode.

Important

Automatic threshold in **ST** mode will be applied to a channel in a well if:

1. no rule is specified for this channel in the related reaction mix

2. the well is not part of any reaction mix specified in the analysis template
3. no rules are specified *at all* for step 1, but some rules are specified for steps 2 or 3

Example

Experiment configuration:

- the experiment involves an imaging step for GREEN, YELLOW and CRIMSON channels
- well A1 and A2 have a reaction mix with a name of RM1
- well A3 and A4 have a reaction mix with a name of RM2
- all other wells do not have a reaction mix specified

Analysis template configuration of step 1:

- for reaction mix with a name of RM1, set the thresholds to the following values:
 - GREEN channel: 100
 - YELLOW channel: 150
- for reaction mix with a name of RM2, set the thresholds to the following values:
 - CRIMSON channel: 800

Once the analysis template is applied, the thresholds will be set as follows:

Well	Reaction mix	GREEN	YELLOW	CRIMSON
A1	RM1	100	150	automatic (ST mode)
A2	RM1	100	150	automatic (ST mode)
A3	RM2	automatic (ST mode)	automatic (ST mode)	800
A4	RM2	automatic (ST mode)	automatic (ST mode)	800
A5	-	automatic (ST mode)	automatic (ST mode)	automatic (ST mode)
A6	-	automatic (ST mode)	automatic (ST mode)	automatic (ST mode)
...	-	automatic (ST mode)	automatic (ST mode)	automatic (ST mode)

9.1.2 Step 2 - validation of the quality of the results

This step validates if the results meet the quality criteria. If the criteria are not met on at least one channel in a well, then such well is marked as failing the validation.

The validation results are used in the next step. Also, they are provided in the response for `GetExperimentResults` endpoint.

9.1.3 Step 3 - setting up dynamic thresholds

This step allows to set the thresholds to a value equal to the arithmetic mean of thresholds from controls or NTCs having the particular reaction mix. If a well did not pass the validation step, it is excluded from the calculation of the mean.

9.2 Identifying rules in the experiment results

The `GetExperimentResults` endpoint includes a `thresholdSource` field for each well and channel which - in case of thresholds set via analysis template - provides the name of the rule that set the threshold. Possible values are:

- `InitialAutoThresholdOverrideRule.ForReactionMix(<reaction_mix_name>)` - the threshold was set by the rule from step 1 for the reaction mix with a particular name, example: `InitialAutoThresholdOverrideRule.ForReactionMix(RM1)`
- `InitialAutoThresholdOverrideRule.NotSpecified()` - no rule was configured for this well and channel
- `DynamicThresholdRule.ForPositiveControl(<reaction_mix_name>)` - the threshold was set by a rule from step 3 for a positive control from a reaction mix with a particular name, example: `DynamicThresholdRule.ForPositiveControl(RM1)`
- `DynamicThresholdRule.ForNegativeControl(<reaction_mix_name>)` - the threshold was set by a rule from step 3 for a negative control from a reaction mix with a particular name, example: `DynamicThresholdRule.ForNegativeControl(RM1)`
- `DynamicThresholdRule.ForNtcRuleSpecific(<reaction_mix_name>)` - the threshold was set by a rule from step 3 for an NTC from a reaction mix with a particular name, example: `DynamicThresholdRule.ForNtcRuleSpecific(RM1)`

9.3 Manual changes to the thresholds

If a manual change is made to any of the thresholds after the analysis template is applied, then such thresholds will no longer indicate in the `thresholdSource` field that they were set by a rule. This state can be reverted by reapplying the analysis template via the user interface of the Software Suite (see user manual for details).

9.4 Automatic reanalysis

Automatic reanalysis will happen when analysis template is already configured on a plate and one of the following happens:

- analysis template is changed;
- name of any reaction mix is changed;
- reaction mix assignment to wells is changed;
- control types are changed;
- control or NTC assignment to wells is changed;

This is done in order to ensure that results are in sync with the analysis template.

10 Changelog

10.1 3.1.1.0

- the `EPXERIMENT_READY` event got an extension of its schema, it now indicates if all imaging steps of an experiment are ready;
- the `DRAWER_CLOSED` event will now be triggered after the instrument finishes identification the plates on the drawer(s). Only then the instrument will proceed with the next command in the queue. Previously, in case `RunExperiment` was the next command in the queue, it could be processed too soon, before the plates were identified and it could result in a `'NO_PLATE'` error.

- the **OpenDrawer** command will wait until the internal robot stops moving before opening the drawer. Previously, this command would result in an error if it was requested while the internal robot was still moving.

10.2 3.2.0.0

- new endpoint added: **GetRfuDetails**
- **AddParametersToExperiment** endpoint:
 - sample ID may now be provided. This allows to create samples with the same name but different IDs. The use case here is setting up different dilution and conversion factors for the same sample (same as in having the same name).
- **AssignAnalysisTemplate** endpoint:
 - allows to set up automatic amplitude multiplexing thresholds;
 - allows to set up “quality assurance” validation rules and dynamic thresholds based on controls or NTCs.
- **GetExperimentResults** endpoint:
 - provides values of the automatic threshold(s) and an **isAutoThreshold** flag which indicates if the current threshold is automatic;
 - provides the **lambda** value in the concentration results;
 - provides a new field **qualityAssuranceValidationRules**. If a validation rule was set up for a well in **AssignAnalysisTemplate**, then this field specifies the type of the rule and whether the validation has passed or not;
 - provides new types of automation rules in **thresholdSource** field; for details, see the documentation chapter for assigning analysis templates.
 - in previous API version, if an automation rule set a threshold to the automatic threshold, then **thresholdSource** field would have a value of { **"type": "Auto"** }. Now, in such case, **thresholdSource** will have a value of { **"type": "AutomationRule", "name": "name of the rule"** }. To easily recognize that auto threshold is set, an additional flag **isAutoThreshold** is present.
- (bugfix) **GetInstrumentStatus** endpoint presents drawer names in the correct letter case. In the previous API version the drawer names were always stated in lowercase. If such drawer name was used in a command, instrument would not recognize it and - as an example - would not open a drawer.

11 Running an experiment – an end-to-end scenario example with real API calls

Prerequisites:

- Lab Automation Service is enabled
- An instrument is registered in Suite (refer to the QIAcuity Software Suite User Manual for instructions)

11.1 Define an experiment

11.1.1 Option A – define an experiment based on an existing template

Create a template of a plate in Software Suite (refer to the QIAcuity Software Suite User Manual for instructions):

```
curl -X 'POST' \
'https://localhost:8687/lab-automation/v1/experiment/define/template' \
```

```
-H 'accept: application/json' \
-H 'Authorization: ApiKey some_api_key' \
-H 'Content-Type: application/json' \
-d '{
  "barcode": "00031234567891113151719212",
  "plateName": "example_plate_name",
  "templateName": "example_template_name",
  "owners": [ "admin" ]
}'
```

Response:

2a7e25f7-c2c2-469e-8edb-3b3b9430d83a

11.1.2 Option B – define an experiment from scratch

First, load dictionaries with plate type IDs:

```
curl -X 'GET' \
  'https://localhost:8687/lab-automation/v1/experiment/plate-types' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key'
```

Response:

```
{
  "bb52fd68-effd-4168-a2a3-8b3e6dfdb597": "Nanoplate 26K 8-well",
  "d27fb9bb-bc7f-4502-9b51-b5fd552b8837": "Nanoplate 26K 24-well"
}
```

Then, request for velocity profiles for a particular type of plate:

```
curl -X 'GET' \
  'https://localhost:8687/lab-automation/v1/experiment/velocity-profiles?plateTypeId=d27fb9bb-bc7f-4502-9b51-b5fd552b8837' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key'
```

Response:

```
{
  "d2844690-b50f-4563-b764-fd32ed95e76b": "QIAGEN Priming Profile Probe (RT-) PCR",
  "97a3e10f-2484-4f88-bf13-ac1911e4423b": "QIAGEN Priming Profile EvaGreen (RT-) PCR"
}
```

Now, define an experiment:

```
curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/experiment/define' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
  -H 'Content-Type: application/json' \
  -d '{
    "barcode": "00011234567891113151719212",
    "plateName": "example_plate_name",
    "plateTypeId": "d27fb9bb-bc7f-4502-9b51-b5fd552b8837",
    "dpcrRunSteps": {
      "primingDPCRStep": {
        "velocityProfile": "d2844690-b50f-4563-b764-fd32ed95e76b"
      },
      "cyclingDPCRSteps": [
        {
          "index": 1,
          "cycles": [
            {
              "count": 1,
              "position": 0,
              "steps": [
                {
                  "temperature": 55,
                  "duration": 55
                }
              ]
            }
          ]
        }
      ],
      "imagingDPCRSteps": [
        {
          "index": 2,
          "referenceChannel": "REF2",
          "imagingProfiles": [
            {
              "channel": {
```

```

        "excitation": "GREEN",
        "emission": "GREEN"
    },
    "durationOfExposure": 4000,
    "gain": 40
}
]
}
],
"owners": [
    "admin"
]
}'

```

Note: the request contains the barcode of the plate. This is optional. If it is provided, the instrument will do a sanity check to recognize if the correct plate was put into a particular slot on the drawer. If it is not provided, the instrument will automatically read the barcode and assign it to the plate once the experiment is run.

Response:

```
2a7e25f7-c2c2-469e-8edb-3b3b9430d83a
```

This is the ID of the plate created in Software Suite.

11.2 Put the plate into the instrument

To operate a particular drawer, a booking needs to be made for it on a particular instrument. To get the list of instruments:

```

curl -X 'GET' \
  'https://localhost:8687/lab-automation/v1/instruments' \
  -H 'accept: application/json'
  -H 'Authorization: ApiKey some_api_key'

```

Response:

```

[
  {
    "instrumentId": "instrument123",
    "deviceName": "Instrument in Lab A",
    "type": "P4",
    "isOnline": true,

```

```

    "drawers": {
      "Drawer0": {
        "isBooked": false,
        "platesInSlots": {
          "0": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
        }
      }
    }
  }
}
]

```

The response tells that this is an instrument with one drawer “Drawer1” and that “slot 0” is occupied by a plate. The drawer is not booked, so the booking can be made:

```

curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/command/drawer/book' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
  -H 'Content-Type: application/json' \
  -d '{
    "instrumentId": "instrument123",
    "drawerName": "Drawer0"
  }'

```

Response:

```
"a392f741-3015-4a6a-883f-42d64cb5159a"
```

This is the command ID. The event telling that the booking is active will be soon available via this endpoint:

```

curl -X 'GET' \
  'https://localhost:8687/lab-automation/v1/event' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key'

```

This endpoint needs to be polled until the event appears in the response:

```

{
  "id": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "commandId": "a392f741-3015-4a6a-883f-42d64cb5159a",
  "instrumentId": "instrument123",
  "type": "DRAWER_BOOKED",
  "event": "{\"freeSlotsInDrawers\":{\"Drawer0\":[1,2,3]}}",

```

```
  "payloadSchemaVersion": 1
}
```

The event must be acknowledged to be able to see the events that follow.

```
curl -X 'DELETE' \
  'https://localhost:8687/lab-automation/v1/event?eventId=21ebe258-ec80-4f73-aafc-4faa863d8a22' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key'
```

Note: different commands will result in different event types and payloads. However, all events need to be acknowledged.

When the booking is done, the drawer can be opened:

```
curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/command/drawer/open' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
  -H 'Content-Type: application/json' \
  -d '{
    "instrumentId": "instrument123",
    "drawerName": "Drawer0"
  }'
```

Wait for the confirmation event and acknowledge it. Then, the plate may be placed on a free slot and the drawer may be closed:

```
curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/command/drawer/close' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
  -H 'Content-Type: application/json' \
  -d '{
    "instrumentId": "instrument123",
    "drawerName": "Drawer0"
  }'
```

11.3 Run the experiment

```
curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/command/experiment/run' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
```

```
-H 'Content-Type: application/json' \
-d '{
  "instrumentId": "instrument123",
  "plateId": "2a7e25f7-c2c2-469e-8edb-3b3b9430d83a",
  "drawerName": "Drawer0",
  "slotId": 1
}'
```

Await the event of type EXPERIMENT_PROCESSING_STARTED.

11.4 [Optionally] Release the booking

Now, you can optionally release the booking of the drawer if you wish to enable human interaction with the instrument. If you have a fully-automated use scenario, you don't need to release the booking at all.

```
curl -X 'POST' \
  'https://localhost:8687/lab-automation/v1/command/drawer/release-booking' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key' \
  -H 'Content-Type: application/json' \
  -d '{
    "instrumentId": "instrument123",
    "drawerName": "Drawer0"
  }'
```

11.5 Wait for the results to be ready

Await the event of type EXPERIMENT_READY. Before it happens, multiple events of type EXPERIMENT_PROGRESS will be sent to inform about the progress of the experiment like the current dPCR step.

Please note that EXPERIMENT_PROGRESS event with a status of RUN_COMPLETED does not mean the results are ready. Until EXPERIMENT_READY is sent, the results are still being processed by the Software Suite.

11.6 Download the results

```
curl -X 'GET' \
  'https://localhost:8687/lab-automation/v1/experiment/2a7e25f7-c2c2-469e-8edb-3b3b9430d83a/result' \
  -H 'accept: application/json' \
  -H 'Authorization: ApiKey some_api_key'
```

12 Endpoints

Interactive API specification (Swagger UI) can be accessed via the browser at https://<URL_TO_SOFTWARE_SUITE>:8687/swagger-ui/index.html. It can also be accessed in form of an OpenAPI specification at https://<URL_TO_SOFTWARE_SUITE>:8687/api-docs/lab-automation-api.

12.1 AddExperimentParameters

PUT /lab-automation/v1/experiment/{plateId}/add-parameters

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds reactions, samples and wells configuration to an existing experiment. The provided parameters will overwrite the already existing parameters, so the request body must contain all desired experiment parameters even if only a single detail (e.g. reaction mix name) is to be changed.

12.1.1 Request body example

```
{
  "reactions": [
    {
      "name": "Reaction1",
      "totalReactionVolume": 30.5,
      "targets": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN",
            "thresholdMode": "ST"
          },
          "name": "Target1",
          "internalControl": false,
          "dyeId": "1ef3cc8a-8a5e-4709-8101-e8293fd63e2d"
        }
      ]
    }
  ],
  "sampleControl": {
    "samples": [
      {
        "name": "Sample1",
        "sampleId": 1,

```

```

        "volumeOfTemplate": 15,
        "concentrationFactor": 15,
        "conversionFactor": 15,
        "conversionUnit": "lm/kg",
        "description": "Description1",
        "labels": [
            "string"
        ]
    },
    "controls": [
        {
            "name": "Control1"
        }
    ],
    "nonTemplateControls": [
        {
            "name": "NonTemplateControl1"
        }
    ]
},
"wells": [
    {
        "wellRow": "A",
        "column": 1,
        "blank": false,
        "reactionName": "Reaction1",
        "sample": {
            "name": "Sample1",
            "sampleId": 1,
            "type": "SAMPLE"
        },
        "controlTypes": [
            {
                "channel": {
                    "excitation": "GREEN",
                    "emission": "GREEN",
                    "thresholdMode": "ST"
                }
            }
        ]
    }
]

```

```

    "positive": true
  }
]
}

```

12.1.1.2 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment for which parameters should be updated
body	body	AddParametersToExperimentRequest	true	Replaces existing parameters with requested ones

12.1.1.3 Example responses

12.1.1.3.1 200 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.1.1.4 Responses

Status	Meaning	Description	Schema
200	OK	Returns ID of updated plate	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.2 AddDpcrSteps

PUT /lab-automation/v1/experiment/{plateId}/add-dpcr-steps

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds cycling and imaging steps to an existing experiment.

The steps must follow these rules:

- a cycling must be followed by an imaging
- imaging step may be followed by another imaging step

Steps may only be added after the already existing steps and they must continue the indexing. There must not be a gap in the indexes.

Example:

An experiment already contains three dPCR steps:

- priming at index 0
- cycling at index 1
- imaging at index 2

The next step must have an index of 3, the next one must have an index of 4 and so on.

12.2.1 Request body example

```
{
  "cyclingDPCRSteps": [
    {
      "index": 3,
      "cycles": [
        {
          "count": 1,
          "position": 0,
          "steps": [
            {
              "temperature": 35,
              "duration": 0,
              "position": 0
            }
          ]
        }
      ]
    }
  ]
},
```

```

"imagingDPCRSteps": [
  {
    "index": 4,
    "enableHighMultiplexingReferenceChannel": true,
    "imagingProfiles": [
      {
        "channel": {
          "excitation": "GREEN",
          "emission": "GREEN"
        },
        "durationOfExposure": 1,
        "gain": 40
      }
    ]
  }
]
}

```

12.2.2 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment to which dPCR steps should be added
body	body	AddDPCRStepsToExperimentRequest	true	Adds requested dPCR steps to existing ones

12.2.3 Example responses

12.2.3.1 200 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.2.4 Responses

Status	Meaning	Description	Schema
200	OK	Returns ID of updated plate	string
400	Bad Request	Missing or invalid requested params	ErrorResponse

Status	Meaning	Description	Schema
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.3 GetRfuDetails

POST /lab-automation/v1/experiment/{plateId}/rfu

To perform this operation, you must be authenticated by means of one of the following methods: **Lab Automation API Key Authorization**.

Provides RFU values for the requested experiment

12.3.1 Request body example

```
{
  "dpcrRunStepIndex": 2,
  "wellPositions": [
    1,
    2,
    4
  ],
  "channels": [
    {
      "excitation": "GREEN",
      "emission": "GREEN"
    }
  ],
  "sortType": "VERTICAL"
}
```

12.3.2 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment

Name	In	Type	Required	Description
body	body	RfuRequest	true	none

12.3.3 Example responses

12.3.3.1 200 Response

null

12.3.4 Responses

Status	Meaning	Description	Schema
200	OK	Get RFU details	StreamingResponseBody
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.4 AssignAnalysisTemplate

POST /lab-automation/v1/experiment/{plateId}/analysis-template

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Assigns an analysis template to the requested experiment and triggers the analysis process if applicable

12.4.1 Request body example

```
{
  "reactionMixAutomationRules": {
    "ReactionMix A": {
      "initialAutoThresholdOverrideRules": [
        {
          "channel": {
            "excitation": "GREEN",
```

```

        "emission": "GREEN"
    },
    "threshold": {
        "threshold": 60.3,
        "type": "ST"
    }
},
{
    "channel": {
        "excitation": "YELLOW",
        "emission": "YELLOW"
    },
    "threshold": {
        "low": 20.2,
        "high": 50.9,
        "double": 70.4,
        "type": "AM"
    }
}
]
},
"ReactionMix B": {
    "initialAutoThresholdOverrideRules": [
        {
            "channel": {
                "excitation": "GREEN",
                "emission": "GREEN"
            },
            "threshold": {
                "threshold": 44.5,
                "type": "ST"
            }
        }
    ],
    "validationRules": {
        "minimalValidPartitionsPercentage": 30
    },
    "dynamicThresholdRules": [
        {

```

```

    "channel": {
      "excitation": "GREEN",
      "emission": "GREEN"
    },
    "rule": {
      "type": "FOR_POSITIVE_CONTROL"
    }
  },
  {
    "channel": {
      "excitation": "YELLOW",
      "emission": "YELLOW"
    },
    "rule": {
      "type": "FOR_NTC",
      "offset": 4
    }
  }
]
}
}
}

```

12.4.2 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment
body	body	AssignAnalysisTemplateRequest	true	none

12.4.3 Example responses

12.4.3.1 400 Response

```

{
  "message": "string",
  "code": "string",
  "uuid": "095be615-a8ad-4c33-8e9c-c7612fbf6c9f",
  "validationErrors": {

```

```

    "property1": [
      {
        "code": "string",
        "arguments": [
          "string"
        ]
      }
    ],
    "property2": [
      {
        "code": "string",
        "arguments": [
          "string"
        ]
      }
    ]
  }
}

```

12.4.4 Responses

Status	Meaning	Description	Schema
200	OK	Analysis template assigned	None
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.5 DefineExperiment

POST /lab-automation/v1/experiment/define

To perform this operation, you must be authenticated by means of one of the following methods: **Lab Automation API Key Authorization**.

Defines an experiment based on provided parameters

12.5.1 Request body example

```
{
  "barcode": "00031234567891113151719212",
  "plateName": "example_plate_name",
  "plateTypeId": "67e02b72-768c-4bde-a4ec-d4c70c1c777a",
  "dpcrRunSteps": {
    "primingDPCRStep": {
      "velocityProfile": "0282a51b-422f-491a-8e66-cebc06cb8990"
    },
    "cyclingDPCRSteps": [
      {
        "index": 1,
        "cycles": [
          {
            "count": 1,
            "position": 0,
            "steps": [
              {
                "temperature": 40,
                "duration": 180,
                "position": 0
              }
            ]
          }
        ]
      }
    ],
    "imagingDPCRSteps": [
      {
        "index": 2,
        "enableHighMultiplexingReferenceChannel": true,
        "imagingProfiles": [
          {
            "channel": {
              "excitation": "GREEN",
              "emission": "GREEN"
            },
            "durationOfExposure": 1,

```

```

    "gain": 40
  }
]
},
"owners": [
  "admin",
  "carl"
]
}

```

12.5.2 Request body properties

Name	In	Type	Required	Description
body	body	DefineExperimentRequest	true	none

12.5.3 Example responses

12.5.3.1 200 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.5.4 Responses

Status	Meaning	Description	Schema
200	OK	Returns ID of created experiment	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.6 DefineExperimentFromTemplate

POST /lab-automation/v1/experiment/define/template

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Defines an experiment based on existing template

12.6.1 Request body example

```
{
  "barcode": "00031234567891113151719212",
  "plateName": "example_plate_name",
  "templateName": "example_template_name",
  "owners": [
    "admin",
    "carl"
  ]
}
```

12.6.2 Request body properties

Name	In	Type	Required	Description
body	body	DefineExperimentFromTemplateRequest	true	none

12.6.3 Example responses

12.6.3.1 200 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.6.4 Responses

Status	Meaning	Description	Schema
200	OK	Returns ID of created experiment (plate ID)	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate template not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.7 RunExperiment

POST /lab-automation/v1/command/experiment/run

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds command that runs a previously defined plate experiment, returns queued command's ID

12.7.1 Request body example

```
{
  "instrumentId": "instrument123",
  "plateId": "21ebe258-ec80-4f73-aafc-4faa863d8a22",
  "drawerName": "Drawer1",
  "slotId": 2
}
```

12.7.2 Request body properties

Name	In	Type	Required	Description
body	body	RunExperimentRequest	true	none

12.7.3 Example responses

12.7.3.1 201 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.7.4 Responses

Status	Meaning	Description	Schema
201	Created	Command successfully added to the queue. Returns queued command's ID	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None

Status	Meaning	Description	Schema
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.8 ReleaseDrawerBooking

POST /lab-automation/v1/command/drawer/release-booking

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds command that releases drawers from exclusive robot use to queue, returns queued command's ID

12.8.1 Request body example

```
{
  "instrumentId": "instrument123",
  "drawerName": "Drawer1"
}
```

12.8.2 Request body properties

Name	In	Type	Required	Description
body	body	ReleaseDrawerBookingRequest	true	none

12.8.3 Example responses

12.8.3.1 201 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.8.4 Responses

Status	Meaning	Description	Schema
201	Created	Command successfully added to the queue. Returns queued command's ID	string

Status	Meaning	Description	Schema
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.9 OpenDrawer

POST /lab-automation/v1/command/drawer/open

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds command that opens a specific drawer which was previously booked, returns queued command's ID

12.9.1 Request body example

```
{
  "instrumentId": "instrument123",
  "drawerName": "Drawer1"
}
```

12.9.2 Request body properties

Name	In	Type	Required	Description
body	body	OpenDrawerRequest	true	none

12.9.3 Example responses

12.9.3.1 201 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.9.4 Responses

Status	Meaning	Description	Schema
201	Created	Command successfully added to the queue. Returns queued command's ID	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.10 CloseDrawer

POST /lab-automation/v1/command/drawer/close

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds command that closes a specific drawer which was previously booked, returns queued command's ID

12.10.1 Request body example

```
{
  "instrumentId": "instrument123",
  "drawerName": "Drawer1"
}
```

12.10.2 Request body properties

Name	In	Type	Required	Description
body	body	CloseDrawerRequest	true	none

12.10.3 Example responses

12.10.3.1 201 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.10.4 Responses

Status	Meaning	Description	Schema
201	Created	Command successfully added to the queue. Returns queued command's ID	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.11 BookDrawer

POST /lab-automation/v1/command/drawer/book

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Adds command that books a drawer for exclusive robot use to queue, returns queued command's ID

12.11.1 Request body example

```
{  
  "instrumentId": "instrument123",  
  "drawerName": "Drawer1"  
}
```

12.11.2 Request body properties

Name	In	Type	Required	Description
body	body	BookDrawerRequest	true	none

12.11.3 Example responses

12.11.3.1 201 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.11.4 Responses

Status	Meaning	Description	Schema
201	Created	Command successfully added to the queue. Returns queued command's ID	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.12 GetInstruments

GET /lab-automation/v1/instruments

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Returns the state of the instruments registered in Suite.

12.12.1 Example responses

12.12.1.1 200 Response

```
[
  {
    "instrumentId": "instrument123",
    "deviceName": "Instrument in Lab A",
    "type": "P4",
    "isOnline": true,
    "drawers": {
      "Drawer0": {
        "isBooked": true,
        "platesInSlots": {}
      }
    }
  },
]
```

```

    "Drawer1": {
      "isBooked": false,
      "platesInSlots": {
        "0": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "2": null
      }
    }
  }
}
]

```

12.12.2 Responses

Status	Meaning	Description	Schema
200	OK	Successfully retrieved the instruments.	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.12.3 Response Schema

12.12.3.1 Status Code 200

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	[InstrumentStateResponse]	false	none	none
» instrumentId	string	false	none	ID of the instrument
» deviceName	string	false	none	Name of the instrument
» type	string	false	none	Type of the instrument
» isOnline	boolean	false	none	Whether an instrument has sent a heartbeat in the last PT5S (ISO8601 notation).
» drawers	object	false	none	Drawer state per drawer name
»» <dictionary>	object	false	none	none

Name	Type	Required	Restrictions	Description
»»» isBooked	boolean	false	none	Whether the drawer is currently booked or not
»»» platesInSlots	object	false	none	A dictionary containing experiment (plate) ID per slot ID. If there is no plate in a particular slot, an entry for this slot will not be present in the dictionary. If there is a plate in the slot and a non-readonly plate with the same barcode is present in the Software Suite, the ID of this plate will be displayed. If such plate does not exist in the Software Suite, the ID will be null.
»»»» <dictionary>	string(uuid)	false	none	none

12.13 HealthCheck

GET /lab-automation/v1/health-check

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Check number of tasks in robot queues per instrument ID

12.13.1 Example responses

12.13.1.1 200 Response

```
{
  "instrumentA": {
    "commandQueueTasks": 4,
    "eventQueueTasks": 2
  },
  "instrumentB": {
```

```

    "commandQueueTasks": 1,
    "eventQueueTasks": 0
  }
}

```

12.13.2 Responses

Status	Meaning	Description	Schema
200	OK	Returns number of tasks in robot queues per instrument ID	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.13.3 Response Schema

12.13.3.1 Status Code 200

Name	Type	Required	Restrictions	Description
» <dictionary>	object	false	none	none
»» commandQueueTasks	integer(int32)	false	none	Number of tasks in command queue
»» eventQueueTasks	integer(int32)	false	none	Number of tasks in event queue

12.14 GetExperimentStatus

GET /lab-automation/v1/experiment/{plateId}/status

To perform this operation, you must be authenticated by means of one of the following methods: **Lab Automation API Key Authorization**.

Fetches status of the most recent run of the requested experiment

12.14.1 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment

12.14.2 Example responses

12.14.2.1 200 Response

```
{
  "status": "IDLE",
  "estimatedTimeTillEndOfExperiment": 300
}
```

12.14.3 Responses

Status	Meaning	Description	Schema
200	OK	Returns the status of the most recent run of the experiment	ExperimentStatusResponse
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.15 GetExperimentResults

GET /lab-automation/v1/experiment/{plateId}/result

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Fetches results for requested experiment

12.15.1 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment which results should be fetched

12.15.2 Example responses

12.15.2.1 200 Response

```
[
  {
    "dpcrRunStepIndex": 2,
    "results": [
      {
        "wellDetails": {
          "wellPosition": 5,
          "rowLetter": "A",
          "columnNumber": 1,
          "replicateWellPositions": [
            5,
            8
          ],
          "cycledVolume": 2.57
        },
        "sample": {
          "name": "S1",
          "uniqueId": 1,
          "volumeOfTemplate": 2,
          "concentrationFactor": 20,
          "type": "SAMPLE",
          "conversionFactor": 10.999999999,
          "conversionUnit": "cp/ml"
        },
        "reaction": {
          "name": "corona_g",
          "totalReactionVolume": 2
        },
        "concentrations": [
```

```

{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN",
    "thresholdMode": "ST"
  },
  "target": {
    "id": "95f6cb04-fc4d-4c03-bd22-30ef11300e8c",
    "name": "COVID-19_G2",
    "controlType": "NONE",
    "isInternalControl": false
  },
  "concentration": {
    "value": 10.484819643300222,
    "concentrationOfOriginalSample": 10.484819643300222,
    "concentrationWithCustomUnitOfDpcrReaction": 0,
    "concentrationWithCustomUnitOfUndilutedSample": 0,
    "dilutedCi": 0,
    "undilutedCi": 0,
    "dilutedConvCi": 0,
    "undilutedConvCi": 0,
    "errorCode": "SATURATION_ERROR",
    "lambda": 5.9643300222
  },
  "validsCount": 23908,
  "positivesCount": 205,
  "negativesCount": 23703,
  "ci": 1.4353845944578065,
  "relativeCi": 0.1369012194096266,
  "meanConcentration": {
    "value": 10.484819643300222,
    "concentrationOfOriginalSample": 10.484819643300222,
    "concentrationWithCustomUnitOfDpcrReaction": 0,
    "concentrationWithCustomUnitOfUndilutedSample": 0,
    "dilutedCi": 0,
    "undilutedCi": 0,
    "dilutedConvCi": 0,
    "undilutedConvCi": 0,
    "errorCode": "SATURATION_ERROR",
  }
}

```

```

        "lambda": 5.9643300222
    },
    "meanRelativeCi": 0.03922849550198258,
    "meanCi": 2.4208812652882585,
    "threshold": [
        7.24,
        8.34,
        9.85
    ],
    "autoThreshold": [
        7.24,
        8.34,
        9.85
    ],
    "isAutoThreshold": true,
    "thresholdSource": {},
    "dispersionMeasure": {
        "stdDevOfUndilutedSample": 299.3133873057806,
        "stdDevInDpcrReaction": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
        "cv": 154.91933384829665
    }
}
],
"qualityAssuranceValidationRules": [
    {
        "settings": 0,
        "passed": true
    }
]
}
]

```

12.15.3 Responses

Status	Meaning	Description	Schema
200	OK	Returns full experiment results	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.15.4 Response Schema

12.15.4.1 Status Code 200

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	[ExperimentResultsResponse]	false	none	none
» dpcrRunStepIndex	integer(int32)	false	none	dPCR run step index
» results	[AbsoluteQuantificationResult]	false	none	List of absolute quantification results for requested plate, run step, channels and wells
»» wellDetails	object	false	none	none
»»» wellPosition	integer(int32)	true	none	Well position
»»» rowLetter	string	false	none	Row letter
»»» columnNumber	integer(int32)	false	none	Column number
»»»	[integer]	false	none	Well positions of replicate wells
replicateWellPositions				
»»» cycledVolume	number(double)	false	none	Cycled volume of this well
»» sample	object	false	none	none
»»» name	string	false	none	Name of the sample
»»» uniqueId	integer(int32)	false	none	Unique id of the sample
»»» volumeOfTemplate	number(double)	false	none	Dilution Factor: volume of template value
»»» concentrationFactor	number(double)	false	none	Dilution Factor: concentration factor value

Name	Type	Required	Restrictions	Description
»»» type	string	false	none	Sample type - SAMPLE, CONTROL or NON_TEM-PLATE_CONTROL
»»» conversionFactor	number	false	none	Conversion Factor
»»» conversionUnit	string	false	none	Unit for conversion factor value
»» reaction	object	false	none	none
»»» name	string	false	none	Reaction mix name
»»» totalReactionVolume	number(double)	false	none	Dilution Factor: total reaction volume value
»» concentrations	[AbsoluteQuantification-Result.ConcentrationPer-ChannelWithMean]	false	none	Concentrations per channel with mean data for single well
»»» channel	object	false	none	none
»»»» excitation	string	true	none	Excitation filter color
»»»» emission	string	true	none	Emission filter color
»»»» thresholdMode	string	true	none	Channel threshold mode
»»» target	object	false	none	none
»»»» id	string(uuid)	false	none	Target ID
»»»» name	string	false	none	Target name
»»»» controlType	string	false	none	Control type (positive, negative or none)
»»»» isInternalControl	boolean	false	none	Internal control indicator
»»» concentration	object	false	none	none
»»»» value	number(double)	false	none	Concentration of diluted sample
»»»» concentrationOfOriginalSample	number(double)	false	none	Concentration of sample before dilution
»»»» concentrationWithCustomUnitOfDpcrReaction	number	false	none	Conversion Factor multiplied by concentration of dpcr reaction result
»»»» concentrationWithCustomUnitOfUndilutedSample	number	false	none	Conversion Factor multiplied by concentration of undiluted Sample result

Name	Type	Required	Restrictions	Description
»»»» dilutedCi	number	false	none	Diluted CI value
»»»» undilutedCi	number	false	none	Undiluted CI value
»»»» dilutedConvCi	number	false	none	Diluted CI value for concentration with custom unit of dpcr reaction
»»»» undilutedConvCi	number	false	none	Undiluted CI value for concentration with custom unit of undiluted sample
»»»» errorCode	string	false	none	Error code
»»»» lambda	number(double)	false	none	Lambda
»»» validsCount	integer(int32)	false	none	Number of valid partitions
»»» positivesCount	integer(int32)	false	none	Number of positive partitions
»»» negativesCount	integer(int32)	false	none	Number of negative partitions
»»» ci	number(double)	false	none	Confidence interval
»»» relativeCi	number(double)	false	none	Relative confidence interval
»»» meanConcentration	object	false	none	none
»»» meanRelativeCi	number(double)	false	none	Mean confidence interval, if the well has replicates or is a part of hyperwell
»»» meanCi	number(double)	false	none	Mean relative confidence interval, if the well has replicates or is a part of hyperwell
»»» threshold	any	false	none	Thresholds values, for AM [low, high, doublePositives], for SingleThreshold[lower]

oneOf

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	any	false	none	Amplitude multiplexing threshold

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	Signal threshold
»»»»»» type	string	true	none	none

and

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	none
»»»»»» low	number(double)	false	none	none
»»»»»» high	number(double)	false	none	none
»»»»»» doublePositives	number(double)	false	none	none

xor

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	any	false	none	Single threshold value

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	Signal threshold

and

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	none
»»»»» value	number(double)	false	none	none

continued

Name	Type	Required	Restrictions	Description
»»» autoThreshold	any	false	none	Auto Thresholds values, for AM [low, high, doublePositives], for Signal[lower]

oneOf

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	any	false	none	Amplitude multiplexing threshold

xor

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	any	false	none	Single threshold value

continued

Name	Type	Required	Restrictions	Description
»»» isAutoThreshold	boolean	false	none	Is auto or manual threshold
»»» thresholdSource	any	false	none	Where this threshold comes from

oneOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	object	false	none	Automatic threshold
»»»»» type	string	true	none	none

xor

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	any	false	none	Threshold set by analysis automation rule

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	Where this threshold comes from
»»»»»» type	string	true	none	none

and

Name	Type	Required	Restrictions	Description
»»»»» <i>anonymous</i>	object	false	none	none
»»»»»» ruleName	string	false	none	none

xor - discriminator: type

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	object	false	none	Lasso threshold
»»»»» type	string	true	none	none

xor - discriminator: type

Name	Type	Required	Restrictions	Description
»»»» <i>anonymous</i>	object	false	none	Manually set threshold
»»»» type	string	true	none	none

continued

Name	Type	Required	Restrictions	Description
»»» dispersionMeasure	object	false	none	none
»»»» stdDevOfUndilutedSample	number(double)	false	none	Standard deviation calculated with Dilution Factor values
»»»» stdDevInDpcrReaction	number(double)	false	none	Standard deviation calculated in dpcr reaction
»»»» stdDevForConcentrationWithCustomUnitOfUndilutedSample	number	false	none	Standard deviation calculated with custom unit values
»»»» stdDevForConcentrationWithCustomUnitInDpcrReaction	number	false	none	Standard deviation calculated with custom unit in dpcr reaction
»»»» cv	number(double)	false	none	Coefficient of Variation
»» qualityAssuranceValidationRules	[oneOf]	false	none	Quality Assurance validation results

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»» <i>anonymous</i>	object	false	none	none
»»»» type	string	true	none	none

and

Name	Type	Required	Restrictions	Description
» » » <i>anonymous</i>	object	false	none	none
» » » » settings	integer(int32)	false	none	none
» » » » passed	boolean	false	none	none

12.15.4.2 Enumerated Values

Property	Value
type	SAMPLE
type	CONTROL
type	NON_TEMPLATE_CONTROL
excitation	GREEN
excitation	YELLOW
excitation	ORANGE
excitation	RED
excitation	CRIMSON
excitation	FARRED
emission	GREEN
emission	YELLOW
emission	ORANGE
emission	RED
emission	CRIMSON
emission	FARRED
thresholdMode	ST
thresholdMode	AM_LOW
thresholdMode	AM_HIGH
controlType	POSITIVE
controlType	NEGATIVE
controlType	NONE
errorCode	SATURATION_ERROR
errorCode	IMAGE_BLUR
errorCode	MISSING_CHANNEL_IN_CXTM

12.16 GetPrimingProfiles

GET /lab-automation/v1/experiment/velocity-profiles

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Gets a dictionary of available priming (velocity) profiles with its ID for requested plate type

12.16.1 Requeset body properties

Name	In	Type	Required	Description
plateTypeId	query	string(uuid)	true	Plate type ID for which velocity profiles dictionary should be provided

12.16.2 Example responses

12.16.2.1 200 Response

```
{
  "d2844690-b50f-4563-b764-fd32ed95e76b": "QIAGEN Priming Profile Probe (RT-) PCR",
  "97a3e10f-2484-4f88-bf13-ac1911e4423b": "QIAGEN Priming Profile EvaGreen (RT-) PCR"
}
```

12.16.3 Responses

Status	Meaning	Description	Schema
200	OK	Provides a dictionary of velocity profile names by their IDs	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.16.4 Response Schema

12.16.4.1 Status Code 200 *Keys are velocity profile IDs, values are names*

Name	Type	Required	Restrictions	Description
» <dictionary>	string	false	none	none

12.17 GetPlateTypes

GET /lab-automation/v1/experiment/plate-types

To perform this operation, you must be authenticated by means of one of the following methods: **Lab Automation API Key Authorization**.

Provides a dictionary of plate type names by their IDs

12.17.1 Example responses

12.17.1.1 200 Response

```
{
  "bb52fd68-effd-4168-a2a3-8b3e6dfdb597": "Nanoplate 26K 8-well",
  "d27fb9bb-bc7f-4502-9b51-b5fd552b8837": "Nanoplate 26K 24-well"
}
```

12.17.2 Responses

Status	Meaning	Description	Schema
200	OK	Returns plate types dictionary	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.17.3 Response Schema

12.17.3.1 Status Code 200 *Keys are plate type IDs, values are names*

Name	Type	Required	Restrictions	Description
» <dictionary>	string	false	none	none

12.18 GetDyes

GET /lab-automation/v1/experiment/dyes

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Gets a dictionary of available dyes with its ID

12.18.1 Example responses

12.18.1.1 200 Response

```
{
  "97ca742b-259c-48c9-8e49-e3c9588281f5": "VIC",
  "53c43d01-6eee-43ea-bb63-d74dd5e39115": "SYBR Green",
  "68b8965e-6bbc-46a7-b16b-2f496151fa05": "FAM",
  "2be8adbe-5875-45da-9702-2144c2c20ed6": "HEX"
}
```

12.18.2 Responses

Status	Meaning	Description	Schema
200	OK	Provides a dictionary of dye names by their IDs	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.18.3 Response Schema

12.18.3.1 Status Code 200

Name	Type	Required	Restrictions	Description
» <dictionary>	string	false	none	none

12.19 GetEvent

GET /lab-automation/v1/event

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Returns the event that occurred as a result of the executed command. Shows the oldest event from the queue. To remove an event from the queue, it must be acknowledged using the DELETE /event endpoint.

12.19.1 Example responses

12.19.1.1 200 Response

```
{
  "id": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "commandId": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "instrumentId": "instrument1234",
  "type": "DRAWER_BOOKED",
  "payloadSchemaVersion": 1,
  "payload": null
}
```

12.19.2 Responses

Status	Meaning	Description	Schema
200	OK	Returns event with its ID, ID of command from which it resulted and event body.	EventResponse
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
404	Not Found	No items found in event queue	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.20 AcknowledgeEvent

DELETE /lab-automation/v1/event

To perform this operation, you must be authenticated by means of one of the following methods: Lab Automation API Key Authorization.

Acknowledge the event, so it can be deleted from the event queue.

12.20.1 Requeset body properties

Name	In	Type	Required	Description
eventId	query	string(uuid)	true	UUID of an event to be deleted from the queue

12.20.2 Example responses

12.20.2.1 400 Response

```
{
  "message": "string",
  "code": "string",
  "uuid": "095be615-a8ad-4c33-8e9c-c7612fbf6c9f",
  "validationErrors": {
    "property1": [
      {
        "code": "string",
        "arguments": [
          "string"
        ]
      }
    ],
    "property2": [
      {
        "code": "string",
        "arguments": [
          "string"
        ]
      }
    ]
  }
}
```

12.20.3 Responses

Status	Meaning	Description	Schema
200	OK	Event successfully removed from the event queue.	None
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

12.21 DeleteExperiment

DELETE /lab-automation/v1/experiment/{plateId}

To perform this operation, you must be authenticated by means of one of the following methods: **Lab Automation API Key Authorization**.

Deletes an existing experiment

12.21.1 Request body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment which should be deleted

12.21.2 Example responses

12.21.2.1 200 Response

"497f6eca-6276-4993-bfeb-53cbbbba6f08"

12.21.3 Responses

Status	Meaning	Description	Schema
200	OK	Returns ID of deleted experiment	string
400	Bad Request	Missing or invalid requested params	ErrorResponse
401	Unauthorized	Unauthorized	None
403	Forbidden	Access denied	None

Status	Meaning	Description	Schema
404	Not Found	Requested plate not found	ErrorResponse
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

13 Endpoint schemas

13.1 AddParametersToExperimentRequest

```
{
  "reactions": [
    {
      "name": "Reaction1",
      "totalReactionVolume": 30.5,
      "targets": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN",
            "thresholdMode": "ST"
          },
          "name": "Target1",
          "internalControl": false,
          "dyeId": "1ef3cc8a-8a5e-4709-8101-e8293fd63e2d"
        }
      ]
    }
  ],
  "sampleControl": {
    "samples": [
      {
        "name": "Sample1",
        "sampleId": 1,
        "volumeOfTemplate": 15,
        "concentrationFactor": 15,
        "conversionFactor": 15,
        "conversionUnit": "lm/kg",
        "description": "Description1",

```

```

        "labels": [
            "string"
        ]
    },
    "controls": [
        {
            "name": "Control1"
        }
    ],
    "nonTemplateControls": [
        {
            "name": "NonTemplateControl1"
        }
    ]
},
"wells": [
    {
        "wellRow": "A",
        "column": 1,
        "blank": false,
        "reactionName": "Reaction1",
        "sample": {
            "name": "Sample1",
            "sampleId": 1,
            "type": "SAMPLE"
        },
        "controlTypes": [
            {
                "channel": {
                    "excitation": "GREEN",
                    "emission": "GREEN",
                    "thresholdMode": "ST"
                },
                "positive": true
            }
        ]
    }
]

```

```
}
```

Name	Type	Required	Restrictions	Description
reactions	[AddParametersToExperimentRequest.Reaction]	false	none	List of defined reaction mixes with assigned targets
sampleControl	AddParametersToExperimentRequest.SampleControl	false	none	Collection of samples, controls and non template controls definitions
wells	[AddParametersToExperimentRequest.Well]	false	none	Collection of well definitions

13.2 AddParametersToExperimentRequest.Reaction

```
{
  "name": "Reaction1",
  "totalReactionVolume": 30.5,
  "targets": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN",
        "thresholdMode": "ST"
      },
      "name": "Target1",
      "internalControl": false,
      "dyeId": "1ef3cc8a-8a5e-4709-8101-e8293fd63e2d"
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
name	string	true	none	Reaction name
totalReactionVolume	number(double)	false	none	Total reaction volume

Name	Type	Required	Restrictions	Description
targets	[AddParametersToExperimentRequest.Reaction.Target]	true	none	List of assigned targets

13.3 AddParametersToExperimentRequest.Reaction.Target

```
{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN",
    "thresholdMode": "ST"
  },
  "name": "Target1",
  "internalControl": false,
  "dyeId": "1ef3cc8a-8a5e-4709-8101-e8293fd63e2d"
}
```

Name	Type	Required	Restrictions	Description
channel	ChannelSliceDefinition	true	none	Target channel
name	string	true	none	Target name
internalControl	boolean	false	none	Indication whether target is an internal control
dyeId	string(uuid)	false	none	Assigned dye ID. Available dyes can be listed via the /experiment/dyes endpoint.

13.4 AddParametersToExperimentRequest.SampleControl

```
{
  "samples": [
    {
      "name": "Sample1",
      "sampleId": 1,
      "volumeOfTemplate": 15,

```

```

    "concentrationFactor": 15,
    "conversionFactor": 15,
    "conversionUnit": "lm/kg",
    "description": "Description1",
    "labels": [
      "string"
    ]
  }
],
"controls": [
  {
    "name": "Control1"
  }
],
"nonTemplateControls": [
  {
    "name": "NonTemplateControl1"
  }
]
}

```

Name	Type	Required	Restrictions	Description
samples	[AddParametersToExperimentRequest.SampleControl.Sample]	false	none	List of sample definitions
controls	[AddParametersToExperimentRequest.SampleControl.Control]	false	none	List of control definitions
nonTemplateControls	[AddParametersToExperimentRequest.SampleControl.NonTemplateControl]	false	none	List of non template control definitions

13.5 AddParametersToExperimentRequest.SampleControl.Control

```

{
  "name": "Control1"
}

```

Name	Type	Required	Restrictions	Description
name	string	true	none	Control name

13.6 AddParametersToExperimentRequest.SampleControl.NonTemplateControl

```
{
  "name": "NonTemplateControl1"
}
```

Name	Type	Required	Restrictions	Description
name	string	true	none	Non template control name

13.7 AddParametersToExperimentRequest.SampleControl.Sample

```
{
  "name": "Sample1",
  "sampleId": 1,
  "volumeOfTemplate": 15,
  "concentrationFactor": 15,
  "conversionFactor": 15,
  "conversionUnit": "lm/kg",
  "description": "Description1",
  "labels": [
    "string"
  ]
}
```

Name	Type	Required	Restrictions	Description
name	string	true	none	Sample name

Name	Type	Required	Restrictions	Description
sampleId	integer(int32)	false	none	Unique sample id, used for distinction between two samples with same names. Software is using the following algorithm to assign sample IDs:* if sample id is not provided and provided sample names are unique, the software will generate an unique id, starting from 1 and incrementing by 1 for each new sample without an id, omitting ids provided in the request* if sample id is provided, the software will use it as an unique id for the sample
volumeOfTemplate	number(double)	false	none	Volume of template
concentrationFactor	number(double)	false	none	Concentration factor
conversionFactor	number	false	none	Conversion factor
conversionUnit	string	false	none	Conversion unit
description	string	false	none	Sample description
labels	[string]	false	none	List of sample labels

13.8 AddParametersToExperimentRequest.Well

```
{
  "wellRow": "A",
  "column": 1,
  "blank": false,
  "reactionName": "Reaction1",
  "sample": {
    "name": "Sample1",
    "sampleId": 1,
    "type": "SAMPLE"
  }
}
```

```

},
"controlTypes": [
  {
    "channel": {
      "excitation": "GREEN",
      "emission": "GREEN",
      "thresholdMode": "ST"
    },
    "positive": true
  }
]
}

```

Name	Type	Required	Restrictions	Description
wellRow	string	true	none	Well row
column	integer(int32)	false	none	Well column
blank	boolean	true	none	If a well is blank, it explicitly states that it does not have a reaction mix or sample assigned. This is optional.
reactionName	string	false	none	Assigned reaction name

Name	Type	Required	Restrictions	Description
sample	AddParametersToExperimentRequest.Well.Sample	false	none	Specifies the sample or control type assigned to the well. Depending on the type of the sample, the software looks it up in the <code>sampleControl</code> array provided in the request:* for type <code>SAMPLE</code> , it'll look for <code>sampleControl.samples</code> with the specified name* for type <code>CONTROL</code> , it'll look for <code>sampleControl.controls</code> with the specified name* for type <code>NON_TEMPLATE_CONTROL</code> , it'll look for <code>sampleControl.nonTemplateControls</code> with the specified name
controlTypes	[AddParametersToExperimentRequest.Well.ControlType]	false	none	List of control types. Only needs to be provided if a control is assigned to the well.

13.9 AddParametersToExperimentRequest.Well.ControlType

```
{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN",
    "thresholdMode": "ST"
  },
  "positive": true
}
```

Name	Type	Required	Restrictions	Description
channel	ChannelSliceDefinition	true	none	Control type channel
positive	boolean	false	none	Indication whether control type is positive

13.10 AddParametersToExperimentRequest.Well.Sample

```
{
  "name": "Sample1",
  "sampleId": 1,
  "type": "SAMPLE"
}
```

Name	Type	Required	Restrictions	Description
name	string	true	none	Sample, control or non template control name
sampleId	integer(int32)	false	none	Unique sample id, used for distinction between two samples with same names. Needs to be provided if the sample name is not unique within samples defined in plate.
type	string	true	none	Sample type

13.10.0.1 Enumerated Values

Property	Value
type	SAMPLE
type	CONTROL
type	NON_TEMPLATE_CONTROL

13.11 ChannelSliceDefinition

```
{  
  "excitation": "GREEN",  
  "emission": "GREEN",  
  "thresholdMode": "ST"  
}
```

Name	Type	Required	Restrictions	Description
excitation	string	true	none	Excitation filter color
emission	string	true	none	Emission filter color
thresholdMode	string	true	none	Channel threshold mode

13.11.0.1 Enumerated Values

Property	Value
excitation	GREEN
excitation	YELLOW
excitation	ORANGE
excitation	RED
excitation	CRIMSON
excitation	FARRED
emission	GREEN
emission	YELLOW
emission	ORANGE
emission	RED
emission	CRIMSON
emission	FARRED
thresholdMode	ST
thresholdMode	AM_LOW
thresholdMode	AM_HIGH

13.12 ErrorResponse

```
{  
  "message": "string",  
  "code": "string",  
}
```

```

"uuid": "095be615-a8ad-4c33-8e9c-c7612fbf6c9f",
"validationErrors": {
  "property1": [
    {
      "code": "string",
      "arguments": [
        "string"
      ]
    }
  ],
  "property2": [
    {
      "code": "string",
      "arguments": [
        "string"
      ]
    }
  ]
}
}

```

Name	Type	Required	Restrictions	Description
message	string	false	none	none
code	string	false	none	none
uuid	string(uuid)	false	none	none
validationErrors	object	false	none	none
» <dictionary>	[ValidationErrorResponse]	false	none	none

13.13 ValidationErrorResponse

```

{
  "code": "string",
  "arguments": [
    "string"
  ]
}

```

Name	Type	Required	Restrictions	Description
code	string	false	none	none
arguments	[string]	false	none	none

13.14 AddDPCRStepsToExperimentRequest

```
{
  "cyclingDPCRSteps": [
    {
      "index": 3,
      "cycles": [
        {
          "count": 1,
          "position": 0,
          "steps": [
            {
              "temperature": 35,
              "duration": 0,
              "position": 0
            }
          ]
        }
      ]
    }
  ],
  "imagingDPCRSteps": [
    {
      "index": 4,
      "enableHighMultiplexingReferenceChannel": true,
      "imagingProfiles": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN"
          },
          "durationOfExposure": 1,
          "gain": 40
        }
      ]
    }
  ]
}
```

```

    ]
  }
]
}

```

Name	Type	Required	Restrictions	Description
cyclingDPCRSteps	[AddDPCRStepsToExperimentRequest.Cycling]	false	none	List of defined cycling steps
imagingDPCRSteps	[AddDPCRStepsToExperimentRequest.Imaging]	true	none	List of defined cycling steps

13.15 AddDPCRStepsToExperimentRequest.Cycling

```

{
  "index": 3,
  "cycles": [
    {
      "count": 1,
      "position": 0,
      "steps": [
        {
          "temperature": 35,
          "duration": 0,
          "position": 0
        }
      ]
    }
  ]
}

```

Name	Type	Required	Restrictions	Description
index	integer(int32)	false	none	Cycling step index
cycles	[AddDPCRStepsToExperimentRequest.Cycling.Cycle]	false	none	Defined cycles

13.16 AddDPCRStepsToExperimentRequest.Cycling.Cycle

```
{
  "count": 1,
  "position": 0,
  "steps": [
    {
      "temperature": 35,
      "duration": 0,
      "position": 0
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
count	integer(int32)	false	none	Number of cycles
position	integer(int32)	true	none	Cycle position in order
steps	[AddDPCRStepsToExperimentRequest.Cycling.Cycle.CycleStep]	false	none	List of steps within the cycle

13.17 AddDPCRStepsToExperimentRequest.Cycling.Cycle.CycleStep

```
{
  "temperature": 35,
  "duration": 0,
  "position": 0
}
```

Name	Type	Required	Restrictions	Description
temperature	number(double)	true	none	Step temperature
duration	integer(int32)	true	none	Step duration in seconds

Name	Type	Required	Restrictions	Description
position	integer(int32)	true	none	Step position. For all cycle steps within a cycle, each cycle step must have a unique position from 1 to N (where N is the number of cycle steps in the cycle).

13.18 AddDPCRStepsToExperimentRequest.Imaging

```
{
  "index": 4,
  "enableHighMultiplexingReferenceChannel": true,
  "imagingProfiles": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
      "durationOfExposure": 1,
      "gain": 40
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
index	integer(int32)	false	none	Imaging step index
enableHighMultiplexingReferenceChannel	boolean	true	none	Enabling High-Multiplexing Reference channel will make additional channels and amplitude multiplexing “mode” available. Please note that this will require usage of the QIAcuity High Multiplex Kit.

Name	Type	Required	Restrictions	Description
imagingProfiles	[AddDPCRStepsToExperimentRequest.Imaging.ImagingProfile]	false	none	Imaging profiles

13.19 AddDPCRStepsToExperimentRequest.Imaging.ImagingProfile

```
{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN"
  },
  "durationOfExposure": 1,
  "gain": 40
}
```

Name	Type	Required	Restrictions	Description
channel	ChannelDefinition	true	none	Channel name
durationOfExposure	integer(int64)	false	none	Duration of exposure value
gain	integer(int32)	false	none	Gain value

13.20 ChannelDefinition

```
{
  "excitation": "GREEN",
  "emission": "GREEN"
}
```

Name	Type	Required	Restrictions	Description
excitation	string	true	none	Excitation filter color
emission	string	true	none	Emission filter color

13.20.0.1 Enumerated Values

Property	Value
excitation	GREEN
excitation	YELLOW
excitation	ORANGE
excitation	RED
excitation	CRIMSON
excitation	FARRED
emission	GREEN
emission	YELLOW
emission	ORANGE
emission	RED
emission	CRIMSON
emission	FARRED

13.21 RfuRequest

```
{
  "dpcrRunStepIndex": 2,
  "wellPositions": [
    1,
    2,
    4
  ],
  "channels": [
    {
      "excitation": "GREEN",
      "emission": "GREEN"
    }
  ],
  "sortType": "VERTICAL"
}
```

Name	Type	Required	Restrictions	Description
dpcrRunStepIndex	integer(int32)	true	none	DPCR run step index
wellPositions	[integer]	false	none	Selected wells
channels	[ChannelDefinition]	true	none	Selected channels
sortType	string	false	none	Vertical or Horizontal result sorting

13.21.0.1 Enumerated Values

Property	Value
sortType	VERTICAL
sortType	HORIZONTAL

13.22 StreamingResponseBody

null

None

13.23 AssignAnalysisTemplateRequest

```
{
  "reactionMixAutomationRules": {
    "ReactionMix A": {
      "initialAutoThresholdOverrideRules": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN"
          },
          "threshold": {
            "threshold": 60.3,
            "type": "ST"
          }
        },
        {
          "channel": {
            "excitation": "YELLOW",
            "emission": "YELLOW"
          },
          "threshold": {
            "low": 20.2,
            "high": 50.9,
            "double": 70.4,
            "type": "AM"
          }
        }
      ]
    }
  }
}
```

```

    }
  ]
},
"ReactionMix B": {
  "initialAutoThresholdOverrideRules": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
      "threshold": {
        "threshold": 44.5,
        "type": "ST"
      }
    }
  ],
  "validationRules": {
    "minimalValidPartitionsPercentage": 30
  },
  "dynamicThresholdRules": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
      "rule": {
        "type": "FOR_POSITIVE_CONTROL"
      }
    },
    {
      "channel": {
        "excitation": "YELLOW",
        "emission": "YELLOW"
      },
      "rule": {
        "type": "FOR_NTC",
        "offset": 4
      }
    }
  ]
}

```

```

    }
  }
}

```

Name	Type	Required	Restrictions	Description
reactionMixAutomation-Rules	object	true	none	A dictionary with keys representing names of reaction mixes and values representing analysis rules for these reaction mixes
» <dictionary>	AssignAnalysisTemplateRequest.ReactionMixAnalysisRules	false	none	Contains a list of rules for automated analysis

13.24 AssignAnalysisTemplateRequest.AmplitudeMultiplexing

```

{
  "low": 30,
  "high": 55,
  "double": 100
}

```

AssignAnalysisTemplateRequest.AmplitudeMultiplexing

Name	Type	Required	Restrictions	Description
AssignAnalysisTemplateRequest.AmplitudeMultiplexing	any	false	none	Amplitude multiplexing threshold specification

allOf - discriminator: AssignAnalysisTemplateRequest.Threshold.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	AssignAnalysisTemplateRequest.Threshold	false	none	Threshold specification

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» low	number(double)	false	none	Low threshold
» high	number(double)	false	none	High threshold
» double	number(double)	false	none	Double positives threshold

13.25 AssignAnalysisTemplateRequest.AmplitudeMultiplexingAutoThreshold

null

AssignAnalysisTemplateRequest.AmplitudeMultiplexingAutoThreshold

Name	Type	Required	Restrictions	Description
AssignAnalysisTemplateRequest.AmplitudeMultiplexingAutoThreshold	AssignAnalysisTemplateRequest.Threshold	false	none	Amplitude multiplexing auto threshold

13.26 AssignAnalysisTemplateRequest.DynamicThresholdRule

```
{
  "rule": {},
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN"
  }
}
```

Contains a list of rules which will override the dynamic thresholds

Name	Type	Required	Restrictions	Description
rule	any	true	none	none

oneOf

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.ForNegativeControl	false	none	For negative Control specification

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.ForNonTemplateControl	false	none	Contains a list of rules which will override the dynamic thresholds for NTC

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.ForPositiveControl	false	none	For positive Control specification

continued

Name	Type	Required	Restrictions	Description
channel	ChannelDefinition	true	none	none

13.27 AssignAnalysisTemplateRequest.ForNegativeControl

null

AssignAnalysisTemplateRequest.ForNegativeControl

Name	Type	Required	Restrictions	Description
AssignAnalysisTemplateRequest.ForNegativeControl	Rule	false	none	For negative Control specification

13.28 AssignAnalysisTemplateRequest.ForNonTemplateControl

```
{  
  "offset": 0.1  
}
```

Contains a list of rules which will override the dynamic thresholds for NTC

allOf - discriminator: Rule.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	Rule	false	none	none

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» offset	number(double)	false	none	none

13.29 AssignAnalysisTemplateRequest.ForPositiveControl

null

AssignAnalysisTemplateRequest.ForPositiveControl

Name	Type	Required	Restrictions	Description
AssignAnalysisTemplateRequest.ForPositiveControl	Rule	false	none	For positive Control specification

13.30 AssignAnalysisTemplateRequest.InitialAutoThresholdOverrideRule

```
{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN"
  },
  "threshold": {
    "low": 30,
    "high": 55,
    "double": 100
  }
}
```

Contains a list of rules which will override the automatic thresholds

Name	Type	Required	Restrictions	Description
channel	ChannelDefinition	true	none	none
threshold	any	true	none	none

oneOf

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.AmplitudeMultiplexing	false	none	Amplitude multiplexing threshold specification

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.AmplitudeMultiplexingAutoThreshold	false	none	Amplitude multiplexing auto threshold

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	AssignAnalysisTemplateRequest.SingleThreshold	false	none	Single threshold specification

13.31 AssignAnalysisTemplateRequest.ReactionMixAnalysisRules

```
{
  "initialAutoThresholdOverrideRules": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
      "threshold": {
        "low": 30,
        "high": 55,
        "double": 100
      }
    }
  ],
  "validationRules": {
    "minimalValidPartitionsPercentage": 30
  },
  "dynamicThresholdRules": [
    {
      "rule": {},
      "channel": {
```

```

        "excitation": "GREEN",
        "emission": "GREEN"
    }
}
]
}

```

Contains a list of rules for automated analysis

Name	Type	Required	Restrictions	Description
initialAutoThresholdOverrideRules	[AssignAnalysisTemplateRequest.InitialAutoThresholdOverrideRule]	true	none	[Contains a list of rules which will override the automatic thresholds]
validationRules	AssignAnalysisTemplateRequest.ValidationRules	false	none	Validation rules for specific Reaction Mix
dynamicThresholdRules	[AssignAnalysisTemplateRequest.DynamicThresholdRule]	false	none	[Contains a list of rules which will override the dynamic thresholds]

13.32 AssignAnalysisTemplateRequest.SingleThreshold

```

{
  "type": "string",
  "threshold": 0.4
}

```

AssignAnalysisTemplateRequest.SingleThreshold

allOf - discriminator: AssignAnalysisTemplateRequest.Threshold.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	AssignAnalysisTemplateRequest.Threshold	false	none	Threshold specification

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» threshold	number(double)	false	none	Threshold

13.33 AssignAnalysisTemplateRequest.Threshold

```
{
  "type": "string"
}
```

Threshold specification

Name	Type	Required	Restrictions	Description
type	string	true	none	none

13.34 AssignAnalysisTemplateRequest.ValidationRules

```
{
  "minimalValidPartitionsPercentage": 30
}
```

Validation rules for specific Reaction Mix

Name	Type	Required	Restrictions	Description
minimalValidPartition- sPercentage	integer(int32)	false	none	Minimal valid partitions percentage

13.35 Rule

```
{
  "type": "string"
}
```

Name	Type	Required	Restrictions	Description
type	string	true	none	none

13.36 DefineExperimentRequest

```
{
  "barcode": "00031234567891113151719212",
  "plateName": "example_plate_name",
  "plateTypeId": "67e02b72-768c-4bde-a4ec-d4c70c1c777a",
  "dpcrRunSteps": {
    "primingDPCRStep": {
      "velocityProfile": "0282a51b-422f-491a-8e66-cebc06cb8990"
    },
    "cyclingDPCRSteps": [
      {
        "index": 1,
        "cycles": [
          {
            "count": 1,
            "position": 0,
            "steps": [
              {
                "temperature": 40,
                "duration": 180,
                "position": 0
              }
            ]
          }
        ]
      }
    ],
    "imagingDPCRSteps": [
      {
        "index": 2,
        "enableHighMultiplexingReferenceChannel": true,
        "imagingProfiles": [
          {
            "channel": {
              "excitation": "GREEN",
              "emission": "GREEN"
            },
            "durationOfExposure": 1,

```

```

    "gain": 40
  }
]
}
],
"owners": [
  "admin",
  "carl"
]
}

```

Name	Type	Required	Restrictions	Description
barcode	string	false	none	Plate barcode. Used to make a sanity check if the plate in the slot is the one that is defined.
plateName	string	true	none	Optional Plate name
plateTypeId	string(uuid)	true	none	Plate type ID. Available plate types can be listed via the <code>/experiment/plate-types</code> endpoint.
dpcrRunSteps	DefineExperimentRequest.DPCRRunSteps	true	none	dPCR run steps including Priming, Cycling or Imaging.
owners	[string]	false	none	Username of plate owners. The owners must exist in the current system.

13.37 DefineExperimentRequest.DPCRRunSteps

```

{
  "primingDPCRStep": {
    "velocityProfile": "0282a51b-422f-491a-8e66-cebc06cb8990"
  },

```

```

"cyclingDPCRSteps": [
  {
    "index": 1,
    "cycles": [
      {
        "count": 1,
        "position": 0,
        "steps": [
          {
            "temperature": 40,
            "duration": 180,
            "position": 0
          }
        ]
      }
    ]
  }
],
"imagingDPCRSteps": [
  {
    "index": 2,
    "enableHighMultiplexingReferenceChannel": true,
    "imagingProfiles": [
      {
        "channel": {
          "excitation": "GREEN",
          "emission": "GREEN"
        },
        "durationOfExposure": 1,
        "gain": 40
      }
    ]
  }
]
}

```

Name	Type	Required	Restrictions	Description
primingDPCRStep	DefineExperimentRequest.DPCRRunSteps.Priming	true	none	Priming step definition
cyclingDPCRSteps	[DefineExperimentRequest.DPCRRunSteps.Cycling]	true	none	List of defined cycling steps
imagingDPCRSteps	[DefineExperimentRequest.DPCRRunSteps.Imaging]	true	none	List of defined cycling steps

13.38 DefineExperimentRequest.DPCRRunSteps.Cycling

```
{
  "index": 1,
  "cycles": [
    {
      "count": 1,
      "position": 0,
      "steps": [
        {
          "temperature": 40,
          "duration": 180,
          "position": 0
        }
      ]
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
index	integer(int32)	false	none	Cycling step index
cycles	[DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle]	false	none	Defined cycles

13.39 DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle

```
{
  "count": 1,
  "position": 0,
  "steps": [
    {
      "temperature": 40,
      "duration": 180,
      "position": 0
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
count	integer(int32)	false	none	Number of cycles
position	integer(int32)	true	none	Cycle position in order
steps	[DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle.CycleStep]	false	none	List of steps within the cycle

13.40 DefineExperimentRequest.DPCRRunSteps.Cycling.Cycle.CycleStep

```
{
  "temperature": 40,
  "duration": 180,
  "position": 0
}
```

Name	Type	Required	Restrictions	Description
temperature	number(double)	false	none	Step temperature
duration	integer(int32)	true	none	Step duration in seconds
position	integer(int32)	true	none	Step position in order

13.41 DefineExperimentRequest.DPCRRunSteps.Imaging

```
{
  "index": 2,
  "enableHighMultiplexingReferenceChannel": true,
  "imagingProfiles": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
      "durationOfExposure": 1,
      "gain": 40
    }
  ]
}
```

Name	Type	Required	Restrictions	Description
index	integer(int32)	false	none	Imaging step index
enableHighMultiplexin- gReferenceChannel	boolean	true	none	Enabling High-Multiplexing Reference channel will make additional channels and amplitude multiplexing “mode” available. Please note that this will require usage of the QIAcuity High Multiplex Kit.
imagingProfiles	[DefineExperimentRe- quest.DPCRRun- Steps.Imaging.Imaging- Profile]	false	none	Imaging profiles

13.42 DefineExperimentRequest.DPCRRunSteps.Imaging.ImagingProfile

```
{
  "channel": {
```

```

    "excitation": "GREEN",
    "emission": "GREEN"
  },
  "durationOfExposure": 1,
  "gain": 40
}

```

Name	Type	Required	Restrictions	Description
channel	ChannelDefinition	true	none	Channel name
durationOfExposure	integer(int64)	false	none	Duration of exposure value
gain	integer(int32)	false	none	Gain value

13.43 DefineExperimentRequest.DPCRRunSteps.Priming

```

{
  "velocityProfile": "0282a51b-422f-491a-8e66-cebc06cb8990"
}

```

Name	Type	Required	Restrictions	Description
velocityProfile	string(uuid)	true	none	ID of velocity profile used in priming. Available velocity profiles for a particular plate type can be listed via the /experiment/velocity-profiles endpoint.

13.44 DefineExperimentFromTemplateRequest

```

{
  "barcode": "00031234567891113151719212",
  "plateName": "example_plate_name",
  "templateName": "example_template_name",
  "owners": [
    "admin",
    "carl"
  ]
}

```

```

    ]
}

```

Name	Type	Required	Restrictions	Description
barcode	string	false	none	Plate barcode. Used to make a sanity check if the plate in the slot is the one that is defined. Optional
plateName	string	false	none	Plate name. Needs to be provided if given template does not specify plate's name. Optional
templateName	string	true	none	Name of the template which will be used to create a plate
owners	[string]	false	none	List of plate owners existing in the current system

13.45 RunExperimentRequest

```

{
  "instrumentId": "instrument123",
  "plateId": "21ebe258-ec80-4f73-aafc-4faa863d8a22",
  "drawerName": "Drawer1",
  "slotId": 2
}

```

Name	Type	Required	Restrictions	Description
instrumentId	string	true	none	ID of the instrument on which the command is to be executed.
plateId	string(uuid)	true	none	ID of the plate which should be run by the instrument

Name	Type	Required	Restrictions	Description
drawerName	string	false	none	Name of the drawer in which the requested plate is placed
slotId	integer(int32)	false	none	ID of the slot within the drawer in which the requested plate is placed

13.46 ReleaseDrawerBookingRequest

```
{
  "instrumentId": "instrument123",
  "drawerName": "Drawer1"
}
```

Name	Type	Required	Restrictions	Description
instrumentId	string	true	none	ID of the instrument on which the command is to be executed.
drawerName	string	true	none	Name of drawer to be released from exclusive robot use

13.47 OpenDrawerRequest

```
{
  "instrumentId": "instrument123",
  "drawerName": "Drawer1"
}
```

Name	Type	Required	Restrictions	Description
instrumentId	string	true	none	ID of the instrument on which the command is to be executed.
drawerName	string	true	none	Name of the drawer to be opened

13.48 CloseDrawerRequest

```
{  
  "instrumentId": "instrument123",  
  "drawerName": "Drawer1"  
}
```

Name	Type	Required	Restrictions	Description
instrumentId	string	true	none	ID of the instrument on which the command is to be executed.
drawerName	string	true	none	Name of the drawer to be opened

13.49 BookDrawerRequest

```
{  
  "instrumentId": "instrument123",  
  "drawerName": "Drawer1"  
}
```

Name	Type	Required	Restrictions	Description
instrumentId	string	true	none	ID of the instrument on which the command is to be executed.
drawerName	string	true	none	Name of drawer to be booked for exclusive robot use

13.50 InstrumentStateResponse

```
{  
  "instrumentId": "instrument123",  
  "deviceName": "Instrument in Lab A",  
  "type": "P4",  
  "isOnline": true,  
  "drawers": {
```

```

    "Drawer0": {
      "isBooked": true,
      "platesInSlots": {}
    },
    "Drawer1": {
      "isBooked": false,
      "platesInSlots": {
        "0": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "2": null
      }
    }
  }
}

```

Name	Type	Required	Restrictions	Description
instrumentId	string	false	none	ID of the instrument
deviceName	string	false	none	Name of the instrument
type	string	false	none	Type of the instrument
isOnline	boolean	false	none	Whether an instrument has sent a heartbeat in the last PT5S (ISO8601 notation).
drawers	object	false	none	Drawer state per drawer name
» <dictionary>	InstrumentStateResponse.DrawerState	false	none	none

13.51 InstrumentStateResponse.DrawerState

```

{
  "isBooked": true,
  "platesInSlots": {
    "property1": "497f6eca-6276-4993-bfeb-53cbbbba6f08",
    "property2": "497f6eca-6276-4993-bfeb-53cbbbba6f08"
  }
}

```

Name	Type	Required	Restrictions	Description
isBooked	boolean	false	none	Whether the drawer is currently booked or not
platesInSlots	object	false	none	A dictionary containing experiment (plate) ID per slot ID. If there is no plate in a particular slot, an entry for this slot will not be present in the dictionary. If there is a plate in the slot and a non-readonly plate with the same barcode is present in the Software Suite, the ID of this plate will be displayed. If such plate does not exist in the Software Suite, the ID will be null.
» <dictionary>	string(uuid)	false	none	none

13.52 HealthCheck

```
{
  "commandQueueTasks": 4,
  "eventQueueTasks": 2
}
```

Name	Type	Required	Restrictions	Description
commandQueueTasks	integer(int32)	false	none	Number of tasks in command queue
eventQueueTasks	integer(int32)	false	none	Number of tasks in event queue

13.53 ExperimentStatusResponse

```
{
  "status": "IDLE",
  "estimatedTimeTillEndOfExperiment": 300
}
```

```
}
```

Name	Type	Required	Restrictions	Description
status	string	false	none	Status of the experiment
estimatedTimeTillEndOfExperiment	integer(int32)	false	none	Estimated time till end of experiment in seconds. Null if status is IDLE

13.53.0.1 Enumerated Values

Property	Value
status	IDLE
status	RUNNING
status	RUN_STOPPED
status	RUN_COMPLETED
status	RUN_FAILED

13.54 AbsoluteQuantificationResult

```
{
  "wellDetails": {
    "wellPosition": 5,
    "rowLetter": "A",
    "columnNumber": 1,
    "replicateWellPositions": [
      5,
      8
    ],
    "cycledVolume": 2.57
  },
  "sample": {
    "name": "S1",
    "uniqueId": 1,
    "volumeOfTemplate": 2,
    "concentrationFactor": 20,
    "type": "SAMPLE",
    "conversionFactor": 10.999999999,
  }
}
```

```

    "conversionUnit": "cp/ml"
  },
  "reaction": {
    "name": "corona_g",
    "totalReactionVolume": 2
  },
  "concentrations": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN",
        "thresholdMode": "ST"
      },
      "target": {
        "id": "95f6cb04-fc4d-4c03-bd22-30ef11300e8c",
        "name": "COVID-19_G2",
        "controlType": "NONE",
        "isInternalControl": false
      },
      "concentration": {
        "value": 10.484819643300222,
        "concentrationOfOriginalSample": 10.484819643300222,
        "concentrationWithCustomUnitOfDpcrReaction": 0,
        "concentrationWithCustomUnitOfUndilutedSample": 0,
        "dilutedCi": 0,
        "undilutedCi": 0,
        "dilutedConvCi": 0,
        "undilutedConvCi": 0,
        "errorCode": "SATURATION_ERROR",
        "lambda": 5.9643300222
      },
      "validsCount": 23908,
      "positivesCount": 205,
      "negativesCount": 23703,
      "ci": 1.4353845944578065,
      "relativeCi": 0.1369012194096266,
      "meanConcentration": {
        "value": 10.484819643300222,
        "concentrationOfOriginalSample": 10.484819643300222,

```

```

        "concentrationWithCustomUnitOfDpcrReaction": 0,
        "concentrationWithCustomUnitOfUndilutedSample": 0,
        "dilutedCi": 0,
        "undilutedCi": 0,
        "dilutedConvCi": 0,
        "undilutedConvCi": 0,
        "errorCode": "SATURATION_ERROR",
        "lambda": 5.9643300222
    },
    "meanRelativeCi": 0.03922849550198258,
    "meanCi": 2.4208812652882585,
    "threshold": [
        7.24,
        8.34,
        9.85
    ],
    "autoThreshold": [
        7.24,
        8.34,
        9.85
    ],
    "isAutoThreshold": true,
    "thresholdSource": {},
    "dispersionMeasure": {
        "stdDevOfUndilutedSample": 299.3133873057806,
        "stdDevInDpcrReaction": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
        "cv": 154.91933384829665
    }
},
],
"qualityAssuranceValidationRules": [
    {
        "settings": 0,
        "passed": true
    }
]
}

```

Name	Type	Required	Restrictions	Description
wellDetails	AbsoluteQuantification-Result.Well	false	none	Well
sample	AbsoluteQuantification-Result.Sample	false	none	Sample
reaction	AbsoluteQuantification-Result.Reaction	false	none	Reaction mix
concentrations	[AbsoluteQuantification-Result.ConcentrationPerChannelWithMean]	false	none	Concentrations per channel with mean data for single well
qualityAssuranceValidationRules	[oneOf]	false	none	Quality Assurance validation results

13.55 AbsoluteQuantificationResult.ConcentrationPerChannelWithMean

```
{
  "channel": {
    "excitation": "GREEN",
    "emission": "GREEN",
    "thresholdMode": "ST"
  },
  "target": {
    "id": "95f6cb04-fc4d-4c03-bd22-30ef11300e8c",
    "name": "COVID-19_G2",
    "controlType": "NONE",
    "isInternalControl": false
  },
  "concentration": {
    "value": 10.484819643300222,
    "concentrationOfOriginalSample": 10.484819643300222,
    "concentrationWithCustomUnitOfDpcrReaction": 0,
    "concentrationWithCustomUnitOfUndilutedSample": 0,
    "dilutedCi": 0,
    "undilutedCi": 0,
    "dilutedConvCi": 0,
    "undilutedConvCi": 0,
    "errorCode": "SATURATION_ERROR",
    "lambda": 5.9643300222
  }
}
```

```

},
"validsCount": 23908,
"positivesCount": 205,
"negativesCount": 23703,
"ci": 1.4353845944578065,
"relativeCi": 0.1369012194096266,
"meanConcentration": {
  "value": 10.484819643300222,
  "concentrationOfOriginalSample": 10.484819643300222,
  "concentrationWithCustomUnitOfDpcrReaction": 0,
  "concentrationWithCustomUnitOfUndilutedSample": 0,
  "dilutedCi": 0,
  "undilutedCi": 0,
  "dilutedConvCi": 0,
  "undilutedConvCi": 0,
  "errorCode": "SATURATION_ERROR",
  "lambda": 5.9643300222
},
"meanRelativeCi": 0.03922849550198258,
"meanCi": 2.4208812652882585,
"threshold": [
  7.24,
  8.34,
  9.85
],
"autoThreshold": [
  7.24,
  8.34,
  9.85
],
"isAutoThreshold": true,
"thresholdSource": {},
"dispersionMeasure": {
  "stdDevOfUndilutedSample": 299.3133873057806,
  "stdDevInDpcrReaction": 299.3133873057806,
  "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
  "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
  "cv": 154.91933384829665
}

```

}

Name	Type	Required	Restrictions	Description
channel	ChannelSliceDefinition	false	none	Imaging channel
target	AbsoluteQuantification- Result.ConcentrationPer- ChannelWithMean.Target	false	none	Specific gene sequence to be detected by a dPCR process
concentration	ConcentrationWithError- Code	false	none	Concentration of diluted sample and before dilution, with analysis error
validsCount	integer(int32)	false	none	Number of valid partitions
positivesCount	integer(int32)	false	none	Number of positive partitions
negativesCount	integer(int32)	false	none	Number of negative partitions
ci	number(double)	false	none	Confidence interval
relativeCi	number(double)	false	none	Relative confidence interval
meanConcentration	ConcentrationWithError- Code	false	none	Mean concentration, if the well has replicates or is a part of hyperwell, with mean error code
meanRelativeCi	number(double)	false	none	Mean confidence interval, if the well has replicates or is a part of hyperwell
meanCi	number(double)	false	none	Mean relative confidence interval, if the well has replicates or is a part of hyperwell
threshold	any	false	none	Thresholds values, for AM [low, high, doublePositives], for SingleThreshold[lower]

oneOf

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	SignalThreshold.AmplitudeMultiplexing	false	none	Amplitude multiplexing threshold

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	SignalThreshold.SingleThreshold	false	none	Single threshold value

continued

Name	Type	Required	Restrictions	Description
autoThreshold	any	false	none	Auto Thresholds values, for AM [low, high, doublePositives], for Signal[lower]

oneOf

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	SignalThreshold.AmplitudeMultiplexing	false	none	Amplitude multiplexing threshold

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	SignalThreshold.SingleThreshold	false	none	Single threshold value

continued

Name	Type	Required	Restrictions	Description
isAutoThreshold	boolean	false	none	Is auto or manual threshold
thresholdSource	any	false	none	Where this threshold comes from

oneOf

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	ThresholdSource.Auto	false	none	Automatic threshold

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	ThresholdSource.AutomationRule	false	none	Threshold set by analysis automation rule

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	ThresholdSource.Lasso	false	none	Lasso threshold

xor

Name	Type	Required	Restrictions	Description
» <i>anonymous</i>	ThresholdSource.Manual	false	none	Manually set threshold

continued

Name	Type	Required	Restrictions	Description
dispersionMeasure	DispersionMeasureResponse	false	none	Dispersion measure details

13.56 AbsoluteQuantificationResult.ConcentrationPerChannelWithMean.Target

```
{
  "id": "95f6cb04-fc4d-4c03-bd22-30ef11300e8c",
  "name": "COVID-19_G2",
  "controlType": "NONE",
  "isInternalControl": false
}
```

Name	Type	Required	Restrictions	Description
id	string(uuid)	false	none	Target ID
name	string	false	none	Target name
controlType	string	false	none	Control type (positive, negative or none)
isInternalControl	boolean	false	none	Internal control indicator

13.56.0.1 Enumerated Values

Property	Value
controlType	POSITIVE
controlType	NEGATIVE
controlType	NONE

13.57 AbsoluteQuantificationResult.Reaction

```
{
  "name": "corona_g",
  "totalReactionVolume": 2
}
```

Name	Type	Required	Restrictions	Description
name	string	false	none	Reaction mix name
totalReactionVolume	number(double)	false	none	Dilution Factor: total reaction volume value

13.58 AbsoluteQuantificationResult.Sample

```
{
  "name": "S1",
  "uniqueId": 1,
  "volumeOfTemplate": 2,
  "concentrationFactor": 20,
  "type": "SAMPLE",
  "conversionFactor": 10.999999999,
  "conversionUnit": "cp/ml"
}
```

Name	Type	Required	Restrictions	Description
name	string	false	none	Name of the sample
uniqueId	integer(int32)	false	none	Unique id of the sample
volumeOfTemplate	number(double)	false	none	Dilution Factor: volume of template value
concentrationFactor	number(double)	false	none	Dilution Factor: concentration factor value
type	string	false	none	Sample type - SAMPLE, CONTROL or NON_TEMPLATE_CONTROL
conversionFactor	number	false	none	Conversion Factor
conversionUnit	string	false	none	Unit for conversion factor value

13.58.0.1 Enumerated Values

Property	Value
type	SAMPLE
type	CONTROL
type	NON_TEMPLATE_CONTROL

13.59 AbsoluteQuantificationResult.Well

```
{
  "wellPosition": 5,
  "rowLetter": "A",
  "columnNumber": 1,
  "replicateWellPositions": [
    5,
    8
  ],
  "cycledVolume": 2.57
}
```

Name	Type	Required	Restrictions	Description
wellPosition	integer(int32)	true	none	Well position
rowLetter	string	false	none	Row letter
columnNumber	integer(int32)	false	none	Column number
replicateWellPositions	[integer]	false	none	Well positions of replicate wells
cycledVolume	number(double)	false	none	Cycled volume of this well

13.60 ConcentrationWithErrorCode

```
{
  "value": 10.484819643300222,
  "concentrationOfOriginalSample": 10.484819643300222,
  "concentrationWithCustomUnitOfDpcrReaction": 0,
  "concentrationWithCustomUnitOfUndilutedSample": 0,
  "dilutedCi": 0,
  "undilutedCi": 0,
  "dilutedConvCi": 0,
  "undilutedConvCi": 0,
  "errorCode": "SATURATION_ERROR",
  "lambda": 5.9643300222
}
```

Name	Type	Required	Restrictions	Description
value	number(double)	false	none	Concentration of diluted sample
concentrationOfOriginal-Sample	number(double)	false	none	Concentration of sample before dilution
concentrationWithCustomUnitOfDpcrReaction	number	false	none	Conversion Factor multiplied by concentration of dpcr reaction result
concentrationWithCustomUnitOfUndilutedSample	number	false	none	Conversion Factor multiplied by concentration of undiluted Sample result
dilutedCi	number	false	none	Diluted CI value
undilutedCi	number	false	none	Undiluted CI value
dilutedConvCi	number	false	none	Diluted CI value for concentration with custom unit of dpcr reaction
undilutedConvCi	number	false	none	Undiluted CI value for concentration with custom unit of undiluted sample
errorCode	string	false	none	Error code
lambda	number(double)	false	none	Lambda

13.60.0.1 Enumerated Values

Property	Value
errorCode	SATURATION_ERROR
errorCode	IMAGE_BLUR
errorCode	MISSING_CHANNEL_IN_CXTM

13.61 DispersionMeasureResponse

```
{
  "stdDevOfUndilutedSample": 299.3133873057806,
  "stdDevInDpcrReaction": 299.3133873057806,
  "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
  "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
  "cv": 154.91933384829665
}
```

Name	Type	Required	Restrictions	Description
stdDevOfUndilutedSample	number(double)	false	none	Standard deviation calculated with Dilution Factor values
stdDevInDpcrReaction	number(double)	false	none	Standard deviation calculated in dpcr reaction
stdDevForConcentrationWithCustomUnitOfUndilutedSample	number	false	none	Standard deviation calculated with custom unit values
stdDevForConcentrationWithCustomUnitInDpcrReaction	number	false	none	Standard deviation calculated with custom unit in dpcr reaction
cv	number(double)	false	none	Coefficient of Variation

13.62 ExperimentResultsResponse

```
{
  "dpcrRunStepIndex": 2,
  "results": [
    {
      "wellDetails": {
        "wellPosition": 5,
        "rowLetter": "A",
        "columnNumber": 1,
        "replicateWellPositions": [
          5,
          8
        ]
      }
    }
  ]
}
```

```

    ],
    "cycledVolume": 2.57
  },
  "sample": {
    "name": "S1",
    "uniqueId": 1,
    "volumeOfTemplate": 2,
    "concentrationFactor": 20,
    "type": "SAMPLE",
    "conversionFactor": 10.999999999,
    "conversionUnit": "cp/ml"
  },
  "reaction": {
    "name": "corona_g",
    "totalReactionVolume": 2
  },
  "concentrations": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN",
        "thresholdMode": "ST"
      },
      "target": {
        "id": "95f6cb04-fc4d-4c03-bd22-30ef11300e8c",
        "name": "COVID-19_G2",
        "controlType": "NONE",
        "isInternalControl": false
      },
      "concentration": {
        "value": 10.484819643300222,
        "concentrationOfOriginalSample": 10.484819643300222,
        "concentrationWithCustomUnitOfDpcrReaction": 0,
        "concentrationWithCustomUnitOfUndilutedSample": 0,
        "dilutedCi": 0,
        "undilutedCi": 0,
        "dilutedConvCi": 0,
        "undilutedConvCi": 0,
        "errorCode": "SATURATION_ERROR",

```

```

    "lambda": 5.9643300222
  },
  "validsCount": 23908,
  "positivesCount": 205,
  "negativesCount": 23703,
  "ci": 1.4353845944578065,
  "relativeCi": 0.1369012194096266,
  "meanConcentration": {
    "value": 10.484819643300222,
    "concentrationOfOriginalSample": 10.484819643300222,
    "concentrationWithCustomUnitOfDpcrReaction": 0,
    "concentrationWithCustomUnitOfUndilutedSample": 0,
    "dilutedCi": 0,
    "undilutedCi": 0,
    "dilutedConvCi": 0,
    "undilutedConvCi": 0,
    "errorCode": "SATURATION_ERROR",
    "lambda": 5.9643300222
  },
  "meanRelativeCi": 0.03922849550198258,
  "meanCi": 2.4208812652882585,
  "threshold": [
    7.24,
    8.34,
    9.85
  ],
  "autoThreshold": [
    7.24,
    8.34,
    9.85
  ],
  "isAutoThreshold": true,
  "thresholdSource": {},
  "dispersionMeasure": {
    "stdDevOfUndilutedSample": 299.3133873057806,
    "stdDevInDpcrReaction": 299.3133873057806,
    "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
    "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
    "cv": 154.91933384829665
  }
}

```

```

    }
  }
],
"qualityAssuranceValidationRules": [
  {
    "settings": 0,
    "passed": true
  }
]
}
]
}

```

Name	Type	Required	Restrictions	Description
dpcrRunStepIndex	integer(int32)	false	none	dPCR run step index
results	[AbsoluteQuantification-Result]	false	none	List of absolute quantification results for requested plate, run step, channels and wells

13.63 MinimumValidPartitions

```

{
  "settings": 0,
  "passed": true
}

```

allOf - discriminator: QualityAssuranceValidationRuleResults.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	QualityAssuranceValidationRuleResults	false	none	none

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» settings	integer(int32)	false	none	none

Name	Type	Required	Restrictions	Description
» passed	boolean	false	none	none

13.64 QualityAssuranceValidationRuleResults

```
{
  "type": "string"
}
```

Name	Type	Required	Restrictions	Description
type	string	true	none	none

13.65 SignalThreshold

```
{
  "type": "string"
}
```

Signal threshold

Name	Type	Required	Restrictions	Description
type	string	true	none	none

13.66 SignalThreshold.AmplitudeMultiplexing

```
{
  "low": 0.1,
  "high": 0.1,
  "doublePositives": 0.1
}
```

SignalThreshold.AmplitudeMultiplexing

Name	Type	Required	Restrictions	Description
SignalThreshold.AmplitudeMultiplexing	any	false	none	Amplitude multiplexing threshold

allOf - discriminator: SignalThreshold.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	SignalThreshold	false	none	Signal threshold

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» low	number(double)	false	none	none
» high	number(double)	false	none	none
» doublePositives	number(double)	false	none	none

13.67 SignalThreshold.SingleThreshold

```
{
  "value": 0.1
}
```

SignalThreshold.SingleThreshold

Name	Type	Required	Restrictions	Description
SignalThreshold.SingleThreshold	any	false	none	Single threshold value

allOf - discriminator: SignalThreshold.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	SignalThreshold	false	none	Signal threshold

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» value	number(double)	false	none	none

13.68 ThresholdSource

```
{  
  "type": "string"  
}
```

Where this threshold comes from

Name	Type	Required	Restrictions	Description
type	string	true	none	none

13.69 ThresholdSource.Auto

null

Automatic threshold

None

13.70 ThresholdSource.AutomationRule

```
{  
  "ruleName": "string"  
}
```

Threshold set by analysis automation rule

allOf - discriminator: ThresholdSource.type

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	ThresholdSource	false	none	Where this threshold comes from

and

Name	Type	Required	Restrictions	Description
<i>anonymous</i>	object	false	none	none
» ruleName	string	false	none	none

13.71 ThresholdSource.Lasso

null

Lasso threshold

None

13.72 ThresholdSource.Manual

null

Manually set threshold

None

13.73 EventResponse

```
{
  "id": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "commandId": "21ebe258-ec80-4f73-aa8c-4faa863d8a22",
  "instrumentId": "instrument1234",
  "type": "DRAWER_BOOKED",
  "payloadSchemaVersion": 1,
  "payload": null
}
```

Name	Type	Required	Restrictions	Description
id	string(uuid)	false	none	ID of the event
commandId	string(uuid)	false	none	ID of the command [optional]
instrumentId	string	false	none	ID of the instrument from which the event originated

Name	Type	Required	Restrictions	Description
type	string	false	none	Event type, used to properly decode the payload
payloadSchemaVersion	integer(int32)	false	none	Payload schema version, used for backward compatibility
payload	JsonNode	false	none	Specifics of the event resulting in command execution by the instrument

13.73.0.1 Enumerated Values

Property	Value
type	DRAWER_BOOKED
type	DRAWER_OPENED
type	DRAWER_NOT_OPENED
type	DRAWER_OPENED_MANUALLY
type	DRAWER_CLOSED
type	DRAWER_NOT_CLOSED
type	DRAWER_CLOSED_MANUALLY
type	DRAWER_BOOKING_RELEASED
type	DRAWER_BOOKING_NOT_RELEASED
type	EXPERIMENT_PROCESSING_STARTED
type	EXPERIMENT_READY
type	EXPERIMENT_PROGRESS
type	EXPERIMENT_ABORTED

13.74 JsonNode

null

None