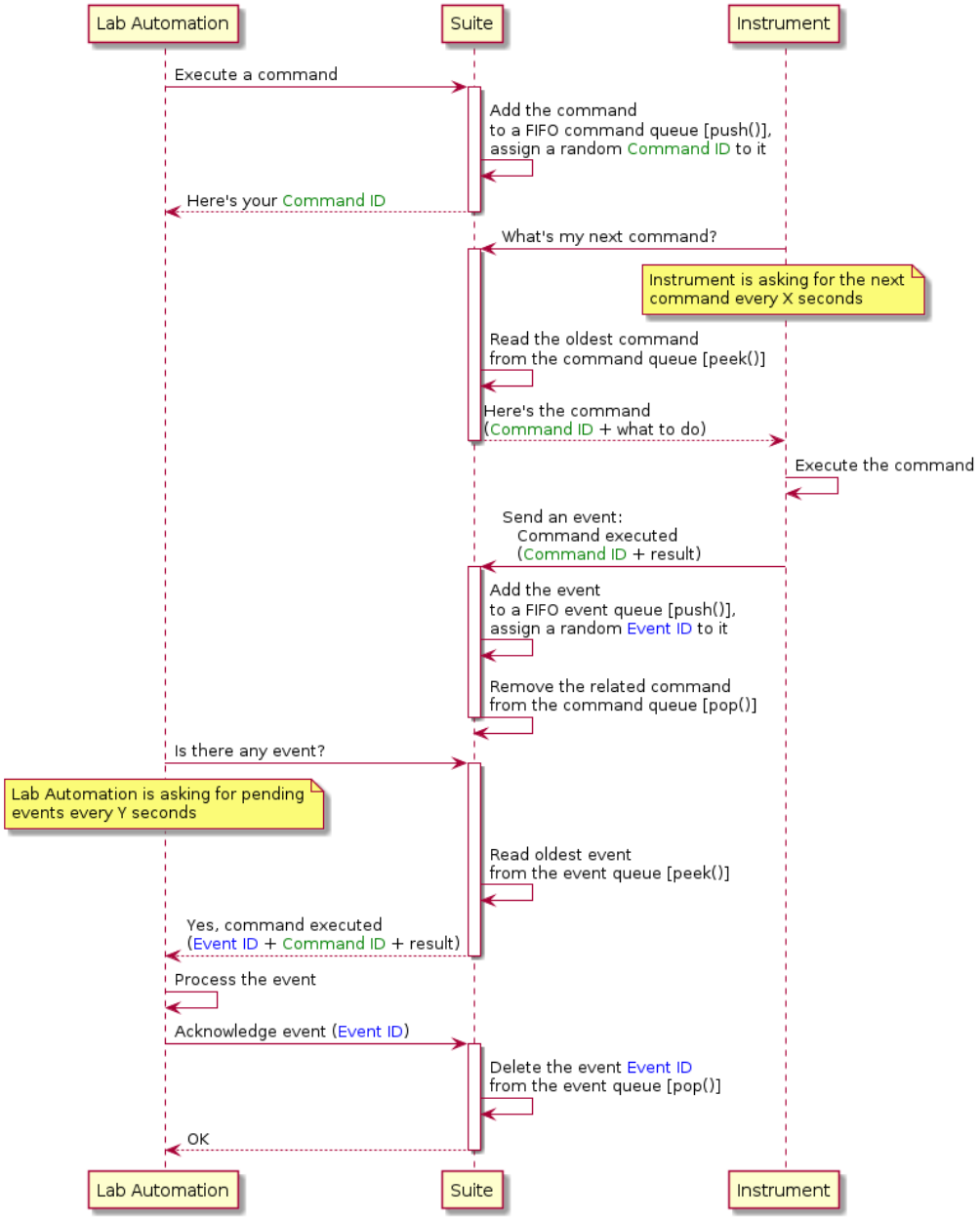
QIAcuity Software Suite API

Contents

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.


Lab Automation
Suite
Instrument
Execute a command
Add the command
to a FIFO command queue [push()],
assign a random Command ID to it
Here's your Command ID
What's my next command?
Instrument is asking for the next
command every X seconds
Read the oldest command
from the command queue [peek()]
Here's the command
(Command ID + what to do)
Execute the command
Send an event:
Command executed
(Command ID + result)
Add the event
to a FIFO event queue [push()],
assign a random Event ID to it
Remove the related command
from the command queue [pop()]
Is there any event?
Lab Automation is asking for pending
events every Y seconds
Read oldest event
from the event queue [peek()]
Yes, command executed
(Event ID + Command ID + result)
Process the event
Acknowledge event (Event ID)
Delete the event Event ID
from the event queue [pop()]
OK
Lab Automation
Suite
Instrument

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 combintations 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.

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.

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-aafc-4faa863d8a22",
  "commandId": "21ebe258-ec80-4f73-aafc-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 and the drawer was successfully closed
- 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 occuring 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 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

Payload:

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

Schema version: 1

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"
```

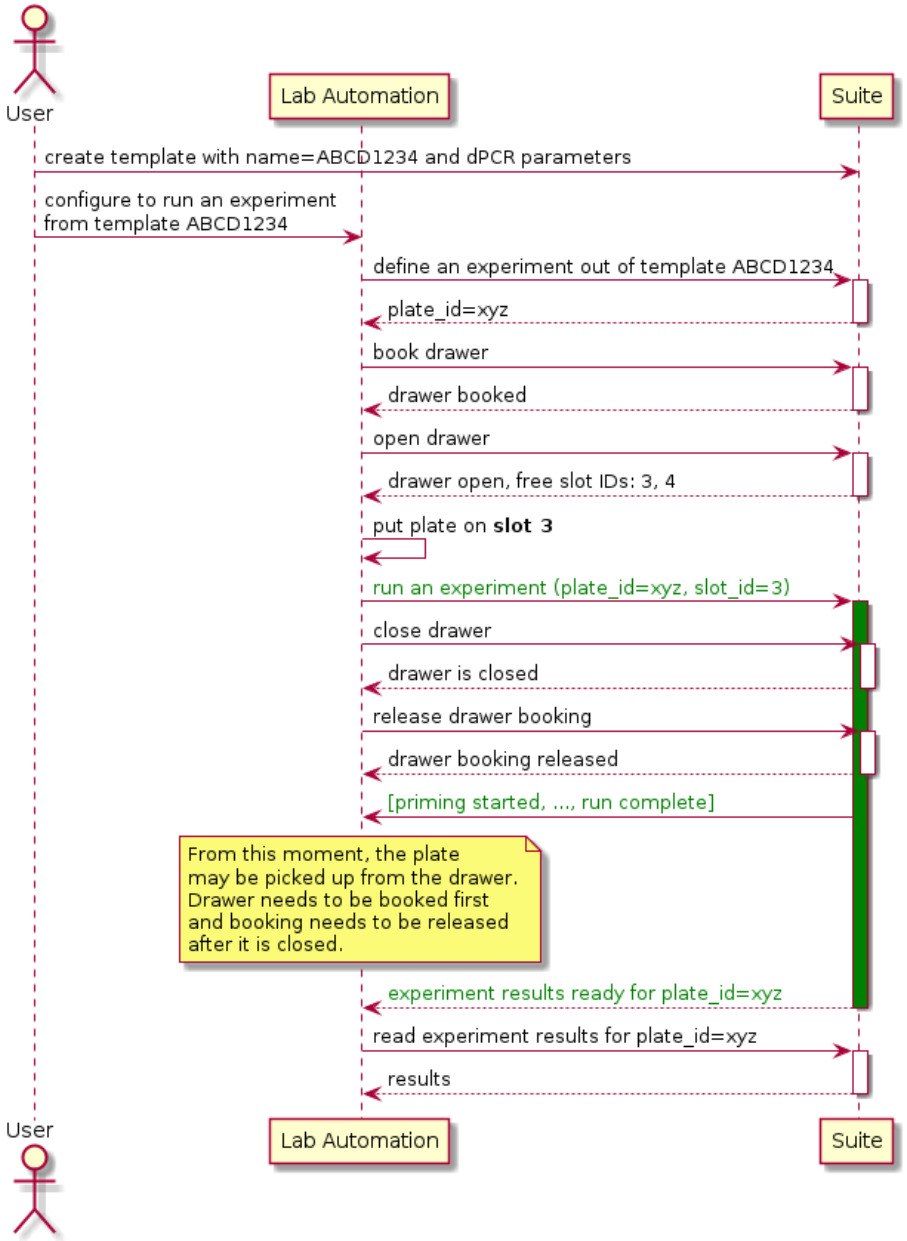
}

`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 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.

9.1 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.

9.1.1 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",
      "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",
        "type": "SAMPLE"
      },
      "controlTypes": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN",
            "thresholdMode": "ST"
          },
          "positive": true
        }
      ]
    }
  ]
}
```


9.1.2 Requset 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

9.1.3 Example responses

9.1.3.1 200 Response


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


9.1.4 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

9.2 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.

9.2.1 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
        }
      ]
    }
  ]
}
```


9.2.2 Requset body properties

Name	In	Type	Required	Description
plateId	path	string(uuid)	true	ID of the experiment to which dPCR steps should be added

Name	In	Type	Required	Description
body	body	AddDPCRStepsToExperimentRequest	true	Adds requested dPCR steps to existing ones

9.2.3 Example responses

9.2.3.1 200 Response

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

9.2.4 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

9.3 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

9.3.1 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,
            "doublePositives": 70.4,
            "type": "AM"
          }
        }
      ]
    },
    "ReactionMix B": {
      "initialAutoThresholdOverrideRules": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN"
          },
          "threshold": {
            "threshold": 44.5,
            "type": "ST"
          }
        }
      ]
    }
  }
}
```

9.3.2 Requset body properties

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

9.3.3 Example responses

9.3.3.1 400 Response


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


9.3.4 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

9.4 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

9.4.1 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"
]
}
```


9.4.2 Requset body properties

Name	In	Type	Required	Description
body	body	DefineExperimentRequest	true	none

9.4.3 Example responses

9.4.3.1 200 Response

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

9.4.4 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

9.5 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

9.5.1 Request body example


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


9.5.2 Requset body properties

Name	In	Type	Required	Description
body	body	DefineExperimentFromTemplateRequest	true	none

9.5.3 Example responses

9.5.3.1 200 Response

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

9.5.4 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
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

9.6 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

9.6.1 Request body example

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

9.6.2 Requset body properties

Name	In	Type	Required	Description
body	body	RunExperimentRequest	true	none

9.6.3 Example responses

9.6.3.1 201 Response

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

9.6.4 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

9.7 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

9.7.1 Request body example

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

9.7.2 Requset body properties

Name	In	Type	Required	Description
body	body	ReleaseDrawerBookingRequest	true	none

9.7.3 Example responses

9.7.3.1 201 Response

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

9.7.4 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
405	Method Not Allowed	HTTP method is not supported	None
500	Internal Server Error	Unexpected error	ErrorResponse

9.8 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

9.8.1 Request body example

```
{
  "instrumentId": "instrument123",
```

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

9.8.2 Requset body properties

Name	In	Type	Required	Description
body	body	OpenDrawerRequest	true	none

9.8.3 Example responses

9.8.3.1 201 Response

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

9.8.4 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

9.9 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

9.9.1 Request body example

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

9.9.2 Requset body properties

Name	In	Type	Required	Description
body	body	CloseDrawerRequest	true	none

9.9.3 Example responses

9.9.3.1 201 Response

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

9.9.4 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

9.10 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

9.10.1 Request body example

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

9.10.2 Requset body properties

Name	In	Type	Required	Description
body	body	BookDrawerRequest	true	none

9.10.3 Example responses

9.10.3.1 201 Response

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

9.10.4 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

9.11 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.

9.11.1 Example responses

9.11.1.1 200 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"
        }
      }
    }
  }
]
```


9.11.2 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

9.11.3 Response Schema

9.11.3.1 Status Code 200

Name	Type	Required	Restrictions	Description
anonymous	[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
»»» isBooked	boolean	false	none	Whether the drawer is currently booked or not
»»» platesInSlots	object	false	none	ID of plate per slot ID.
»»»» **<dictionary>**	string(uuid)	false	none	none

9.12 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

9.12.1 Example responses

9.12.1.1 200 Response


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

```
  "instrumentB": {
    "commandQueueTasks": 1,
    "eventQueueTasks": 0
  }
}
```

9.12.2 Responses

Status	Meaning	Description	Schema
200	OK	Returns number of tasks in robot queues per instrument	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

9.12.3 Response Schema

9.12.3.1 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

9.13 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

9.13.1 Requset body properties

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

9.13.2 Example responses

9.13.2.1 200 Response


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


9.13.3 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

9.14 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

9.14.1 Requset body properties

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

9.14.2 Example responses

9.14.2.1 200 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"
  },
  "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"
  },
```

```
            "meanRelativeCi": 0.03922849550198258,
            "meanCi": 2.4208812652882585,
            "threshold": [
              7.24,
              8.34,
              9.85
            ],
            "thresholdSource": {},
            "dispersionMeasure": {
              "stdDevOfUndilutedSample": 299.3133873057806,
              "stdDevInDpcrReaction": 299.3133873057806,
              "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
              "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
              "cv": 154.91933384829665
            }
          }
        ]
      }
    ]
  }
]
```


9.14.3 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

9.14.4 Response Schema

9.14.4.1 Status Code 200

Name	Type	Required	Restrictions	Description
anonymous	[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
»»» replicateWellPositions	[integer]	false	none	Well positions of replicate wells
»»» 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
»»» 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
»» reaction	object	false	none	none
»»» name	string	false	none	Reaction mix name
»»» totalReactionVolume	number(double)	false	none	Dilution Factor: total reaction volume value
»» concentrations	[AbsoluteQuantificationResult.ConcentrationPerChannelWithMean]	false	none	Concentrations per channel with mean data for single well

Name	Type	Required	Restrictions	Description
»»» 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 dcpr 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
»»» validsCount	integer(int32)	false	none	Number of valid partitions

Name	Type	Required	Restrictions	Description
»»» 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 Signal[lower]

oneOf

Name	Type	Required	Restrictions	Description
»»»» *anonymous*	any	false	none	Amplitude multiplexing threshold

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»» *anonymous*	object	false	none	Signal threshold
»»»»»» type	string	true	none	none

and

Name	Type	Required	Restrictions	Description
»»»»»» *anonymous*	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
»»»»» *anonymous*	any	false	none	Single threshold value

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»»» *anonymous*	object	false	none	Signal threshold

and

Name	Type	Required	Restrictions	Description
»»»»»» *anonymous*	object	false	none	none
»»»»»»» value	number(double)	false	none	none

continued

Name	Type	Required	Restrictions	Description
»»» thresholdSource	any	false	none	Where this threshold comes from

oneOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»» *anonymous*	object	false	none	Automatic threshold
»»»»» type	string	true	none	none

xor

Name	Type	Required	Restrictions	Description
»»»» *anonymous*	any	false	none	Threshold set by analysis automation rule

allOf - discriminator: type

Name	Type	Required	Restrictions	Description
»»»»» *anonymous*	object	false	none	Where this threshold comes from
»»»»»» type	string	true	none	none

and

Name	Type	Required	Restrictions	Description
»»»»» *anonymous*	object	false	none	none
»»»»»» ruleName	string	false	none	none

xor - discriminator: type

Name	Type	Required	Restrictions	Description
»»»» *anonymous*	object	false	none	Manually set threshold
»»»»» type	string	true	none	none

continued

Name	Type	Required	Restrictions	Description
»»» dispersionMeasure	object	false	none	none
»»»» stdDevOfUndiluted-Sample	number(double)	false	none	Standard deviation calculated with Dilution Factor values
»»»» stdDevInDpcrReaction	number(double)	false	none	Standard deviation calculated in dpcr reaction
»»»» stdDevForConcen-trationWithCustomU-nitOfUndilutedSample	number	false	none	Standard deviation calculated with custom unit values
»»»» stdDevForConcen-trationWithCustomU-nitInDpcrReaction	number	false	none	Standard deviation calculated with custom unit in dpcr reaction
»»»» cv	number(double)	false	none	Coefficient of Variation

9.14.4.2 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

Property	Value
controlType	POSITIVE
controlType	NEGATIVE
controlType	NONE
errorCode	SATURATION_ERROR
errorCode	IMAGE_BLUR
errorCode	MISSING_CHANNEL_IN_CXTM

9.15 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

9.15.1 Requset body properties

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

9.15.2 Example responses

9.15.2.1 200 Response


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


9.15.3 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

9.15.4 Response Schema

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

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

9.16 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

9.16.1 Example responses

9.16.1.1 200 Response


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


9.16.2 Responses

Status	Meaning	Description	Schema
200	OK	Returns plate types dictionary	Inline
400	Bad Request	Missing or invalid requested params	ErrorResponse

Status	Meaning	Description	Schema
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

9.16.3 Response Schema

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

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

9.17 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

9.17.1 Example responses

9.17.1.1 200 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"
}
```


9.17.2 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

Status	Meaning	Description	Schema
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

9.17.3 Response Schema

9.17.3.1 Status Code 200

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

9.18 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.

9.18.1 Example responses

9.18.1.1 200 Response


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


9.18.2 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

9.19 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.

9.19.1 Requset body properties

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

9.19.2 Example responses

9.19.2.1 400 Response


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

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


9.19.3 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

9.20 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

9.20.1 Requset body properties

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

9.20.2 Example responses

9.20.2.1 200 Response

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

9.20.3 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
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

10 Endpoint schemas

10.1 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",
      "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",
```

```
        "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

10.2 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
targets	[AddParametersToExperimentRequest.Reaction.Target]	true	none	List of assigned targets

10.3 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

Name	Type	Required	Restrictions	Description
dyeId	string(uuid)	false	none	Assigned dye ID. Available dyes can be listed via the /experiment/dyes endpoint.

10.4 AddParametersToExperimentRequest.SampleControl


```
{
  "samples": [
    {
      "name": "Sample1",
      "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

10.5 AddParametersToExperimentRequest.SampleControl.Control


```
{
  "name": "Control1"
}
```


Name	Type	Required	Restrictions	Description
name	string	true	none	Control name

10.6 AddParametersToExperimentRequest.SampleControl.NonTemplateControl


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


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

10.7 AddParametersToExperimentRequest.SampleControl.Sample


```
{
  "name": "Sample1",
  "volumeOfTemplate": 15,
```

```
  "concentrationFactor": 15,
  "conversionFactor": 15,
  "conversionUnit": "lm/kg",
  "description": "Description1",
  "labels": [
    "string"
  ]
}
```

Name	Type	Required	Restrictions	Description
name	string	true	none	Sample name
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

10.8 AddParametersToExperimentRequest.Well

```
{
  "wellRow": "A",
  "column": 1,
  "blank": false,
  "reactionName": "Reaction1",
  "sample": {
    "name": "Sample1",
    "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
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 `sampleControl` array provided in the request:* for type `SAMPLE`, it'll look for `sampleControl.samples` with the specified name* for type `CONTROL`, it'll look for `sampleControl.controls` with the specified name* for type `NON_TEMPLATE_CONTROL`, it'll look for `sampleControl.nonTemplateControls` 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.

10.9 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

10.10 AddParametersToExperimentRequest.Well.Sample


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


Name	Type	Required	Restrictions	Description
name	string	true	none	Sample, control or non template control name
type	string	true	none	Sample type

10.10.0.1 Enumerated Values

Property	Value
type	SAMPLE
type	CONTROL
type	NON_TEMPLATE_CONTROL

10.11 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

10.11.0.1 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

10.12 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

10.13 ValidationErrorResponse


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

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

10.14 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

10.15 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

10.16 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

10.17 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).

10.18 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

10.19 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

10.20 ChannelDefinition

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

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

10.20.0.1 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

10.21 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,
```

```
            "doublePositives": 70.4,
            "type": "AM"
          }
        }
      ]
    },
    "ReactionMix B": {
      "initialAutoThresholdOverrideRules": [
        {
          "channel": {
            "excitation": "GREEN",
            "emission": "GREEN"
          },
          "threshold": {
            "threshold": 44.5,
            "type": "ST"
          }
        }
      ]
    }
  }
}
```

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>**	AssignAnalysisTem-plateRequest.Reaction-MixAnalysisRules	false	none	Contains a list of rules for automated analysis

10.22 AssignAnalysisTemplateRequest.AmplitudeMultiplexing

```
{
  "double": 100,
```

```
  "low": 30,
  "high": 55,
  "type": "AM"
}
```


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
anonymous	AssignAnalysisTemplateRequest.Threshold	false	none	Threshold specification

and

Name	Type	Required	Restrictions	Description
anonymous	object	false	none	none
» double	number(double)	false	none	Double positives threshold
» low	number(double)	false	none	Low threshold
» high	number(double)	false	none	High threshold
» type	string	false	none	none

10.23 AssignAnalysisTemplateRequest.InitialAutoThresholdOverrideRule


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

```
    "double": 100,
    "low": 30,
    "high": 55,
    "type": "AM"
  }
}
```

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
» *anonymous*	AssignAnalysisTemplateRequest.AmplitudeMultiplexing	false	none	Amplitude multiplexing threshold specification

xor

Name	Type	Required	Restrictions	Description
» *anonymous*	AssignAnalysisTemplateRequest.SingleThreshold	false	none	Single threshold specification

10.24 AssignAnalysisTemplateRequest.ReactionMixAnalysisRules

```
{
  "initialAutoThresholdOverrideRules": [
    {
      "channel": {
        "excitation": "GREEN",
        "emission": "GREEN"
      },
```

```
      "threshold": {
        "double": 100,
        "low": 30,
        "high": 55,
        "type": "AM"
      }
    }
  ]
}
```

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]

10.25 AssignAnalysisTemplateRequest.SingleThreshold

```
{
  "threshold": 0.4,
  "type": "ST"
}
```

AssignAnalysisTemplateRequest.SingleThreshold

Name	Type	Required	Restrictions	Description
AssignAnalysisTemplateRequest.SingleThreshold	any	false	none	Single threshold specification

allOf - discriminator: AssignAnalysisTemplateRequest.Threshold.type

Name	Type	Required	Restrictions	Description
anonymous	AssignAnalysisTemplateRequest.Threshold	false	none	Threshold specification

and

Name	Type	Required	Restrictions	Description
anonymous	object	false	none	none
» threshold	number(double)	false	none	Threshold
» type	string	false	none	none

10.26 AssignAnalysisTemplateRequest.Threshold

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

Threshold specification

Name	Type	Required	Restrictions	Description
type	string	true	none	none

10.27 DefineExperimentRequest

```
{
  "barcode": "000312345678911131517192l2",
  "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. Optional
plateName	string	true	none	Plate name
plateTypeId	string(uuid)	true	none	Plate type ID. Available plate types can be listed via the /experiment/plate-types endpoint.
dpcrRunSteps	DefineExperimentRequest.DPCRRunSteps	true	none	dPCR run steps including Priming, Cycling or Imaging.
owners	[string]	false	none	Usernames of plate owners. The owners must exist in the current system.

10.28 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

10.29 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

10.30 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

10.31 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

10.32 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
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.
imagingProfiles	[DefineExperimentRequest.DPCRRunSteps.Imaging.ImagingProfile]	false	none	Imaging profiles

10.33 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

10.34 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.

10.35 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

Name	Type	Required	Restrictions	Description
owners	[string]	false	none	List of plate owners existing in the current system

10.36 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
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

10.37 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

10.38 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

10.39 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

10.40 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

10.41 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"
      }
    }
  }
}
```

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

10.42 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	ID of plate per slot ID.
» **<dictionary>**	string(uuid)	false	none	none

10.43 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

10.44 ExperimentStatusResponse

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

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

10.44.0.1 Enumerated Values

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

10.45 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"
      },
      "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"
      },
      "meanRelativeCi": 0.03922849550198258,
      "meanCi": 2.4208812652882585,
      "threshold": [
        7.24,
        8.34,
        9.85
      ],
      "thresholdSource": {},
      "dispersionMeasure": {
        "stdDevOfUndilutedSample": 299.3133873057806,
        "stdDevInDpcrReaction": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
        "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
        "cv": 154.91933384829665
      }
    }
  ]
}
```

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.ConcentrationPer-ChannelWithMean]	false	none	Concentrations per channel with mean data for single well

10.46 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"
  },
  "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"
  },
  "meanRelativeCi": 0.03922849550198258,
  "meanCi": 2.4208812652882585,
  "threshold": [
    7.24,
    8.34,
    9.85
  ],
  "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.Tar-get	false	none	Specific gene sequence to be detected by a dPCR process

Name	Type	Required	Restrictions	Description
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 Signal[lower]

oneOf

Name	Type	Required	Restrictions	Description
» *anonymous*	SignalThreshold.Ampli-tudeMultiplexing	false	none	Amplitude multiplexing threshold

xor

Name	Type	Required	Restrictions	Description
» *anonymous*	SignalThreshold.SingleThreshold	false	none	Single threshold value

continued

Name	Type	Required	Restrictions	Description
thresholdSource	any	false	none	Where this threshold comes from

oneOf

Name	Type	Required	Restrictions	Description
» *anonymous*	ThresholdSource.Auto	false	none	Automatic threshold

xor

Name	Type	Required	Restrictions	Description
» *anonymous*	ThresholdSource.AutomationRule	false	none	Threshold set by analysis automation rule

xor

Name	Type	Required	Restrictions	Description
» *anonymous*	ThresholdSource.Manual	false	none	Manually set threshold

continued

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

10.47 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

10.47.0.1 Enumerated Values

Property	Value
controlType	POSITIVE
controlType	NEGATIVE
controlType	NONE

10.48 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

10.49 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

10.49.0.1 Enumerated Values

Property	Value
type	SAMPLE
type	CONTROL
type	NON_TEMPLATE_CONTROL

10.50 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

10.51 ConcentrationWithErrorCode


```
{
  "value": 10.484819643300222,
  "concentrationOfOriginalSample": 10.484819643300222,
  "concentrationWithCustomUnitOfDpcrReaction": 0,
  "concentrationWithCustomUnitOfUndilutedSample": 0,
  "dilutedCi": 0,
  "undilutedCi": 0,
  "dilutedConvCi": 0,
```

```
  "undilutedConvCi": 0,
  "errorCode": "SATURATION_ERROR"
}
```


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
concentrationWithCusto-mUnitOfDpcrReaction	number	false	none	Conversion Factor multiplied by concentration of dcpr reaction result
concentrationWithCusto-mUnitOfUndilutedSam-ple	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

10.51.0.1 Enumerated Values

Property	Value
errorCode	SATURATION_ERROR
errorCode	IMAGE_BLUR
errorCode	MISSING_CHANNEL_IN_CXTM

10.52 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

10.53 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"
```

```
          },
          "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"
          },
          "meanRelativeCi": 0.03922849550198258,
          "meanCi": 2.4208812652882585,
          "threshold": [
            7.24,
            8.34,
            9.85
          ],
          "thresholdSource": {},
          "dispersionMeasure": {
            "stdDevOfUndilutedSample": 299.3133873057806,
            "stdDevInDpcrReaction": 299.3133873057806,
            "stdDevForConcentrationWithCustomUnitOfUndilutedSample": 299.3133873057806,
            "stdDevForConcentrationWithCustomUnitInDpcrReaction": 299.3133873057806,
            "cv": 154.91933384829665
          }
        }
      ]
    }
  ]
}
```

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

10.54 SignalThreshold


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


Signal threshold

Name	Type	Required	Restrictions	Description
type	string	true	none	none

10.55 SignalThreshold.AmplitudeMultiplexing


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


SignalThreshold.AmplitudeMultiplexing

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

allOf - discriminator: SignalThreshold.type

Name	Type	Required	Restrictions	Description
anonymous	SignalThreshold	false	none	Signal threshold

and

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

10.56 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
anonymous	SignalThreshold	false	none	Signal threshold

and

Name	Type	Required	Restrictions	Description
anonymous	object	false	none	none
» value	number(double)	false	none	none

10.57 ThresholdSource


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


Where this threshold comes from

Name	Type	Required	Restrictions	Description
type	string	true	none	none

10.58 ThresholdSource.Auto


```
null
```


Automatic threshold

None

10.59 ThresholdSource.AutomationRule


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


Threshold set by analysis automation rule

allOf - discriminator: ThresholdSource.type

Name	Type	Required	Restrictions	Description
anonymous	ThresholdSource	false	none	Where this threshold comes from

and

Name	Type	Required	Restrictions	Description
anonymous	object	false	none	none
» ruleName	string	false	none	none

10.60 ThresholdSource.Manual

```
null
```

Manually set threshold

None

10.61 EventResponse


```
{
  "id": "21ebe258-ec80-4f73-aafc-4faa863d8a22",
  "commandId": "21ebe258-ec80-4f73-aafc-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
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

10.61.0.1 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

10.62 JsonNode

`null`

None

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-aafc-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"
  }'
```

Once the drawer is closed, the booking can be released:

```
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"
  }'
```

11.3 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_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.

11.4 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'
```