

HPC REST API v1.0.0

Scroll down for example requests and responses.

Limitations

- Maximum number of concurrent connections is 10 per system. When number of concurrent connections exceeds the limit, API responds with status 500 (InternalServerError).

Base URLs:

- <https://jp-east-1.hpc-api.api.nifcloud.com>

Authentication

- API Key (SigV4)
 - Parameter Name: **Authorization**, in: header.

jobs

jobs_get

GET /jobs

Get Job List

Get Job List

This API allows an user to get a list of jobs is still running.

This API is available for authenticated users.

Parameters

| Name | In | Type | Required | Description |
|------------|--------|--------|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |

Example responses

200 Response

```
[
  {
    "job_id": "6291500",
    "job_name": "job1.sh",
```

```
"status": "RNA",
"user": "a03010",
"group": "rccs-aot",
"start_date": "2021-04-12T11:25:00",
"elapsed_time": "0000:00:00",
"node_require": "12",
"priority": "127",
"accept": "2021-04-12T11:25:50",
"queue": "small",
"Fugaku:elapsed_limit": "0000:01:00",
"Fugaku:md": "NM",
"Fugaku:vnode": "-",
"Fugaku:core": "-",
"Fugaku:v_mem": "-",
"Fugaku:v_pol": "-",
"Fugaku:e_pol": "-",
"Fugaku:rank": "bychip",
"Fugaku:lst": "QUE",
"Fugaku:ec": "0",
"Fugaku:pc": "0",
"Fugaku:sn": "0",
"Fugaku:reason": "-"
},
{
  "job_id": "6291501",
  "job_name": "job1.sh",
  "status": "RNA",
  "user": "a03010",
  "group": "rccs-aot",
  "start_date": "2021-04-12T11:25:00",
  "elapsed_time": "0000:00:00",
  "node_require": "12",
  "priority": "127",
  "accept": "2021-04-12T11:25:51",
  "queue": "small",
  "Fugaku:elapsed_limit": "0000:01:00",
  "Fugaku:md": "NM",
  "Fugaku:vnode": "-",
  "Fugaku:core": "-",
  "Fugaku:v_mem": "-",
  "Fugaku:v_pol": "-",
  "Fugaku:e_pol": "-",
  "Fugaku:rank": "bychip",
  "Fugaku:lst": "QUE",
  "Fugaku:ec": "0",
  "Fugaku:pc": "0",
  "Fugaku:sn": "0",
  "Fugaku:reason": "-"
}
]
```

```
{
  "code": "GeneralError",
  "message": "Operation failed."
}
```

Unauthorized

```
{
  "code": "Client.AuthFailurePermission",
  "message": "Authentication API abended."
}
```

```
{
  "code": "Client.AuthFailureRequired.AccessKey",
  "message": "No access key was provided."
}
```

```
{
  "code": "Client.AuthFailureRequired.Signature",
  "message": "No signature was provided."
}
```

```
{
  "code": "Client.AuthFailureExpired",
  "message": "The signature for this request is expired."
}
```

403 Response

```
{
  "code": "Client.AuthFailurePermission.MultiAccount",
  "message": "You don't have enough rights to operate this API."
}
```

404 Response

```
{
  "code": "Client.URLNotFound",
}
```

```

  "message": "A specified URL is not found."
}

```

405 Response

```

{
  "code": "Client.InvalidParameterNotAllowed.Method",
  "message": "A specified HTTP method is not allowed."
}

```

500 Response

```

{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}

```

Service Unavailable

```

{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again
later."
}

```

```

{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}

```

Responses

| Status | Meaning | Description | Schema |
|--------|------------------------------|---------------------|-----------------------------------|
| 200 | OK | Successful Response | Inline |
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |

| Status | Meaning | Description | Schema |
|--------|---------------------------------------|-----------------------|-----------------------------------|
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

Response Schema

Status Code **200**

| Name | Type | Required | Restrictions | Description |
|-----------------------|---|----------|--------------|-------------|
| <i>anonymous</i> | [jobs_get_200_response_inner] | false | none | none |
| » job_id | string | true | none | none |
| » job_name | string | true | none | none |
| » status | string | true | none | none |
| » user | string | true | none | none |
| » group | string | true | none | none |
| » start_date | string | true | none | none |
| » elapse_time | string | false | none | none |
| » node_require | string | true | none | none |
| » priority | string | true | none | none |
| » accept | string | true | none | none |
| » queue | string | true | none | none |
| » Fugaku:elapse_limit | string | true | none | none |
| » Fugaku:md | string | true | none | none |
| » Fugaku:vnode | string | true | none | none |
| » Fugaku:core | string | true | none | none |
| » Fugaku:v_mem | string | true | none | none |
| » Fugaku:v_pol | string | true | none | none |
| » Fugaku:e_pol | string | true | none | none |
| » Fugaku:rank | string | true | none | none |
| » Fugaku:lst | string | true | none | none |
| » Fugaku:ec | string | true | none | none |
| » Fugaku:pc | string | true | none | none |

| Name | Type | Required | Restrictions | Description |
|-----------------|--------|----------|--------------|-------------|
| » Fugaku:sn | string | true | none | none |
| » Fugaku:reason | string | true | none | none |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

jobs_post

POST /jobs

Submit a Job

Submit a Job

This API allows an user to submit a job. Post body consists as following:

```
{
  "jobfile": "scripts/job1.sh",
  "qopt": "-v"
}
```

Job script file is given by `jobfile`. `qopt` is reserved for future extension.

On Successful job commitment, job resource is returned with `Location` header.

This API is available for authenticated users.

Body parameter

```
{
  "jobfile": "string",
  "qopt": "string"
}
```

Parameters

| Name | In | Type | Required | Description |
|--------------|--------|-----------------------------------|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |
| Content-Type | header | string | true | application/json;charset=utf-8 |
| body | body | jobs_post_request | true | none |

Enumerated Values

| Parameter | Value |
|--------------|--------------------------------|
| Content-Type | application/json;charset=utf-8 |

Example responses

201 Response

```
{
  "job_id": 6291500
}
```

General Error

```
{
  "code": "GeneralError",
  "message": "Operation failed."
}
```

```
{
  "code": "GeneralError",
  "message": "Invalid Content-Type."
}
```

```
{
  "code": "GeneralError",
  "message": "Invalid Request Body."
}
```

Unauthorized

```
{
  "code": "Client.AuthFailurePermission",
  "message": "Authentication API abended."
}
```

```
{
  "code": "Client.AuthFailureRequired.AccessKey",
  "message": "No access key was provided."
}
```

```
{
  "code": "Client.AuthFailureRequired.Signature",
  "message": "No signature was provided."
}
```

```
{
  "code": "Client.AuthFailureExpired",
  "message": "The signature for this request is expired."
}
```

403 Response

```
{
  "code": "Client.AuthFailurePermission.MultiAccount",
  "message": "You don't have enough rights to operate this API."
}
```

404 Response

```
{
  "code": "Client.URLNotFound",
  "message": "A specified URL is not found."
}
```

405 Response

```
{
  "code": "Client.InvalidParameterNotAllowed.Method",
  "message": "A specified HTTP method is not allowed."
}
```

422 Response

```
{
  "detail": [
    {
      "loc": [
        "string"
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```



```
]
}
```

500 Response

```
{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}
```

Service Unavailable

```
{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again
later."
}
```

```
{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}
```

Responses

| Status | Meaning | Description | Schema |
|--------|---------------------------------------|-----------------------|--|
| 201 | Created | Successful Response | jobs_post_201_response |
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 422 | Unprocessable Entity | Validation Error | HTTPValidationError |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

Response Headers

| Status | Header | Type | Format | Description |
|--------|----------|--------|--------|---|
| 201 | Location | string | | On Successful job commitment, job resource is returned with Location header. |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

jobs_jobid_delete

DELETE /jobs/{jobid}

Cancel a Job

Cancel a Job

This API allows an user to cancel a running job.

This API is available for authenticated users.

Parameters

| Name | In | Type | Required | Description |
|------------|--------|---------|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |
| jobid | path | integer | true | job_id. |

Example responses

400 Response

```
{
  "code": "GeneralError",
  "message": "Operation Failed."
}
```

Unauthorized

```
{
  "code": "Client.AuthFailurePermission",
  "message": "Authentication API abended."
}
```

```
{
  "code": "Client.AuthFailureRequired.AccessKey",
  "message": "No access key was provided."
}
```

```
{
  "code": "Client.AuthFailureRequired.Signature",
  "message": "No signature was provided."
}
```

```
{
  "code": "Client.AuthFailureExpired",
  "message": "The signature for this request is expired."
}
```

403 Response

```
{
  "code": "Client.AuthFailurePermission.MultiAccount",
  "message": "You don't have enough rights to operate this API."
}
```

Not Found

```
{
  "code": "Client.URLNotFound",
  "message": "A specified URL is not found."
}
```

```
{
  "code": "GeneralError",
  "message": "Operation not permitted."
}
```

405 Response

```
{
  "code": "Client.InvalidParameterNotAllowed.Method",
  "message": "A specified HTTP method is not allowed."
}
```

500 Response

```
{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}
```

Service Unavailable

```
{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again
later."
}
```

```
{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}
```

Responses

| Status | Meaning | Description | Schema |
|--------|---------------------------------------|-----------------------|-----------------------------------|
| 204 | No Content | Successful Response | None |
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

jobs_jobid_get

GET /jobs/{jobid}

Get Job Detail

Get Job Detail

This API allows an user to view a running job's detail.

This API is available for authenticated users.

Parameters

| Name | In | Type | Required | Description |
|------------|--------|---------|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |
| jobid | path | integer | true | job_id. |

Example responses

200 Response

```
{
  "Fugaku:sn": "0",
  "Fugaku:vnode": "-",
  "Fugaku:ec": "0",
  "Fugaku:reason": "-",
  "Fugaku:v_mem": "-",
  "priority": "127",
  "accept": "2021-04-12T11:25:50",
  "Fugaku:rank": "bychip",
  "Fugaku:core": "-",
  "Fugaku:v_pol": "-",
  "Fugaku:lst": "QUE",
  "Fugaku:e_pol": "-",
  "job_name": "job1.sh",
  "job_id": "6291500",
  "elapsed_time": "0000:00:00",
  "node_require": "12",
  "Fugaku:pc": "0",
  "user": "a03010",
  "queue": "small",
  "status": "RNA",
  "group": "rccs-aot",
  "start_date": "2021-04-12T11:25:00",
  "Fugaku:elapsed_limit": "0000:01:00",
  "Fugaku:md": "NM"
}
```

400 Response

```
{
  "code": "GeneralError",
  "message": "Operation Failed."
}
```

Unauthorized

```
{  
  "code": "Client.AuthFailurePermission",  
  "message": "Authentication API abended."  
}
```

```
{  
  "code": "Client.AuthFailureRequired.AccessKey",  
  "message": "No access key was provided."  
}
```

```
{  
  "code": "Client.AuthFailureRequired.Signature",  
  "message": "No signature was provided."  
}
```

```
{  
  "code": "Client.AuthFailureExpired",  
  "message": "The signature for this request is expired."  
}
```

403 Response

```
{  
  "code": "Client.AuthFailurePermission.MultiAccount",  
  "message": "You don't have enough rights to operate this API."  
}
```

Not Found

```
{  
  "code": "Client.URLNotFound",  
  "message": "A specified URL is not found."  
}
```

```
{  
  "code": "GeneralError",  
}
```

```
"message": "Operation not permitted."
}
```

405 Response

```
{
  "code": "Client.InvalidParameterNotAllowed.Method",
  "message": "A specified HTTP method is not allowed."
}
```

500 Response

```
{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}
```

Service Unavailable

```
{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again
later."
}
```

```
{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}
```

Responses

| Status | Meaning | Description | Schema |
|--------|--------------|---------------------|---|
| 200 | OK | Successful Response | jobs_jobid_get_200_response |
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |

| Status | Meaning | Description | Schema |
|--------|---------------------------------------|-----------------------|-----------------------------------|
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

sync files

files_sync_post

POST /files/sync

Start to sync files

Start to sync Files

This API allows an user to sync files. Post body consists as following:

```
{
  "path": "path/to/directory",
  "direction": "from_online",
  "delete": false
}
```

Directory path to synchronized on API Server is given by **path**.

Direction of sync between **API Sever Storage** and **Online Object Storage** is given by **direction** (default is **from_online**). Bucket Name on **Online Object Storage** is predefined fixed value.

- **to_online**: **API Server Storage** is source, and **Online Object Storage** is destination.
- **from_online**: **Online Object Storage** is source, and **API Server Storage** is destination.

If **true** is given by **delete** (default is **false**), files that exist in the destination but not in the source are deleted during sync.

This API is available for authenticated users.

Body parameter

```
{
  "path": "string",
  "direction": "to_online",
  "delete": false
}
```


Parameters

| Name | In | Type | Required | Description |
|--------------|--------|---|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |
| Content-Type | header | string | true | application/json;charset=utf-8 |
| body | body | files_sync_post_request | true | none |

Enumerated Values

| Parameter | Value |
|--------------|--------------------------------|
| Content-Type | application/json;charset=utf-8 |

Example responses

200 Response

```
{
  "requestid": 123456
}
```

General Error

```
{
  "code": "GeneralError",
  "message": "Operation failed."
}
```

```
{
  "code": "GeneralError",
  "message": "Invalid Content-Type."
}
```

```
{
  "code": "GeneralError",
  "message": "Invalid Request Body."
}
```

Unauthorized

```
{  
  "code": "Client.AuthFailurePermission",  
  "message": "Authentication API abended."  
}
```

```
{  
  "code": "Client.AuthFailureRequired.AccessKey",  
  "message": "No access key was provided."  
}
```

```
{  
  "code": "Client.AuthFailureRequired.Signature",  
  "message": "No signature was provided."  
}
```

```
{  
  "code": "Client.AuthFailureExpired",  
  "message": "The signature for this request is expired."  
}
```

403 Response

```
{  
  "code": "Client.AuthFailurePermission.MultiAccount",  
  "message": "You don't have enough rights to operate this API."  
}
```

404 Response

```
{  
  "code": "Client.URLNotFound",  
  "message": "A specified URL is not found."  
}
```

405 Response

```
{  
  "code": "Client.InvalidParameterNotAllowed.Method",
```

```

    "message": "A specified HTTP method is not allowed."
  }

```

422 Response

```

{
  "detail": [
    {
      "loc": [
        "string"
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}

```

500 Response

```

{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}

```

Service Unavailable

```

{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again
later."
}

```

```

{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}

```

Responses

| Status | Meaning | Description | Schema |
|--------|---------|---------------------|--|
| 200 | OK | Successful Response | files_sync_post_200_response |

| Status | Meaning | Description | Schema |
|--------|---------------------------------------|-----------------------|-------------------------------------|
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 422 | Unprocessable Entity | Validation Error | HTTPValidationError |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

files_sync_requestid_get

`GET /files/sync/{requestid}`

Get Synchronization Status

Get Synchronization Status

This API allows an user to view a status of synchronization. Example of response as following:

```
{
  "status": "complete",
  "result": "0"
}
```

Sync Status is set in `status`. If error, message is added after `error`:

- `waiting`
- `complete`
- `error: parameter error`
- `error: non such ID`

Status code is set in `result` only if `status` is `complete`. If sync is successful, 0 is returned. If sync is failed, non-zero value is returned.

This API is available for authenticated users.

Parameters

| Name | In | Type | Required | Description |
|------------|--------|--------|----------|----------------------------------|
| X-Amz-Date | header | string | true | YYYYMMDD'T'HHMMSS'Z' (ISO 8601). |

| Name | In | Type | Required | Description |
|-----------|------|---------|----------|-------------|
| requestid | path | integer | true | request_id. |

Example responses

200 Response

```
{
  "result": "0",
  "status": "complete"
}
```

400 Response

```
{
  "code": "GeneralError",
  "message": "Operation Failed."
}
```

Unauthorized

```
{
  "code": "Client.AuthFailurePermission",
  "message": "Authentication API abended."
}
```

```
{
  "code": "Client.AuthFailureRequired.AccessKey",
  "message": "No access key was provided."
}
```

```
{
  "code": "Client.AuthFailureRequired.Signature",
  "message": "No signature was provided."
}
```

```
{
  "code": "Client.AuthFailureExpired",
  "message": "The signature for this request is expired."
}
```

403 Response

```
{
  "code": "Client.AuthFailurePermission.MultiAccount",
  "message": "You don't have enough rights to operate this API."
}
```

Not Found

```
{
  "code": "Client.URLNotFound",
  "message": "A specified URL is not found."
}
```

```
{
  "code": "GeneralError",
  "message": "Operation not permitted."
}
```

405 Response

```
{
  "code": "Client.InvalidParameterNotAllowed.Method",
  "message": "A specified HTTP method is not allowed."
}
```

500 Response

```
{
  "code": "Server.InternalServerError",
  "message": "An internal error has occurred. Please try your query again
at a later time."
}
```

Service Unavailable

```
{
  "code": "Server.ServiceUnavailable",
  "message": "This service is temporarily unavailable. Please try again"
}
```

```
later."
}
```

```
{
  "code": "Server.Maintenance",
  "message": "The system is in maintenance now."
}
```

Responses

| Status | Meaning | Description | Schema |
|--------|-----------------------|-----------------------|---|
| 200 | OK | Successful Response | files_sync_requestid_get_200_response |
| 400 | Bad Request | General Error | CommonErrorFormat |
| 401 | Unauthorized | Unauthorized | CommonErrorFormat |
| 403 | Forbidden | Forbidden | CommonErrorFormat |
| 404 | Not Found | Not Found | CommonErrorFormat |
| 405 | Method Not Allowed | Method Not Allowed | CommonErrorFormat |
| 500 | Internal Server Error | Internal Server Error | CommonErrorFormat |
| 503 | Service Unavailable | Service Unavailable | CommonErrorFormat |

To perform this operation, you must be authenticated by means of one of the following methods: SigV4

Schemas

CommonErrorFormat

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

CommonErrorFormat

Properties

| Name | Type | Required | Restrictions | Description |
|---------|--------|----------|--------------|-------------|
| code | string | false | none | none |
| message | string | false | none | none |

HTTPValidationError

```
{
  "detail": [
    {
      "loc": [
        "string"
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

HTTPValidationError

Properties

| Name | Type | Required | Restrictions | Description |
|--------|-------------------|----------|--------------|-------------|
| detail | [ValidationError] | false | none | none |

ValidationError

```
{
  "loc": [
    "string"
  ],
  "msg": "string",
  "type": "string"
}
```

ValidationError

Properties

| Name | Type | Required | Restrictions | Description |
|------|----------|----------|--------------|-------------|
| loc | [string] | true | none | none |
| msg | string | true | none | none |
| type | string | true | none | none |

jobs_get_200_response_inner


```

{
  "Fugaku:sn": "Fugaku:sn",
  "Fugaku:vnnode": "Fugaku:vnnode",
  "Fugaku:ec": "Fugaku:ec",
  "Fugaku:reason": "Fugaku:reason",
  "node_require": "node_require",
  "Fugaku:v_mem": "Fugaku:v_mem",
  "priority": "priority",
  "accept": "accept",
  "Fugaku:rank": "Fugaku:rank",
  "Fugaku:core": "Fugaku:core",
  "Fugaku:v_pol": "Fugaku:v_pol",
  "Fugaku:lst": "Fugaku:lst",
  "Fugaku:e_pol": "Fugaku:e_pol",
  "job_name": "job_name",
  "job_id": "job_id",
  "elapse_time": "elapse_time",
  "Fugaku:pc": "Fugaku:pc",
  "user": "user",
  "queue": "queue",
  "status": "status",
  "group": "group",
  "start_date": "start_date",
  "Fugaku:elapse_limit": "Fugaku:elapse_limit",
  "Fugaku:md": "Fugaku:md"
}

```

Properties

| Name | Type | Required | Restrictions | Description |
|--------------|--------|----------|--------------|-------------|
| job_id | string | true | none | none |
| job_name | string | true | none | none |
| status | string | true | none | none |
| user | string | true | none | none |
| group | string | true | none | none |
| start_date | string | true | none | none |
| elapse_time | string | false | none | none |
| node_require | string | true | none | none |
| priority | string | true | none | none |
| accept | string | true | none | none |
| queue | string | true | none | none |

| Name | Type | Required | Restrictions | Description |
|---------------------|--------|----------|--------------|-------------|
| Fugaku:elapse_limit | string | true | none | none |
| Fugaku:md | string | true | none | none |
| Fugaku:vnode | string | true | none | none |
| Fugaku:core | string | true | none | none |
| Fugaku:v_mem | string | true | none | none |
| Fugaku:v_pol | string | true | none | none |
| Fugaku:e_pol | string | true | none | none |
| Fugaku:rank | string | true | none | none |
| Fugaku:lst | string | true | none | none |
| Fugaku:ec | string | true | none | none |
| Fugaku:pc | string | true | none | none |
| Fugaku:sn | string | true | none | none |
| Fugaku:reason | string | true | none | none |

jobs_post_request

```
{
  "jobfile": "string",
  "qopt": "string"
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|---------|--------|----------|--------------|---|
| jobfile | string | true | none | Job script file is given by <code>jobfile</code> . |
| qopt | string | false | none | <code>qopt</code> is reserved for future extension. |

jobs_post_201_response

```
{
  "job_id": 0
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|--------|---------|----------|--------------|-------------|
| job_id | integer | true | none | job id |

jobs_jobid_get_200_response

```
{
  "Fugaku:sn": "Fugaku:sn",
  "Fugaku:vnode": "Fugaku:vnode",
  "Fugaku:ec": "Fugaku:ec",
  "Fugaku:reason": "Fugaku:reason",
  "Fugaku:v_mem": "Fugaku:v_mem",
  "priority": "priority",
  "accept": "accept",
  "Fugaku:rank": "Fugaku:rank",
  "Fugaku:core": "Fugaku:core",
  "Fugaku:v_pol": "Fugaku:v_pol",
  "Fugaku:lst": "Fugaku:lst",
  "Fugaku:e_pol": "Fugaku:e_pol",
  "job_name": "job_name",
  "job_id": "job_id",
  "elapsed_time": "elapsed_time",
  "Fugaku:pc": "Fugaku:pc",
  "user": "user",
  "queue": "queue",
  "status": "status",
  "group": "group",
  "start_date": "start_date",
  "Fugaku:elapsed_limit": "Fugaku:elapsed_limit",
  "Fugaku:md": "Fugaku:md"
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|--------------|--------|----------|--------------|-------------|
| job_id | string | true | none | none |
| job_name | string | true | none | none |
| status | string | true | none | none |
| user | string | true | none | none |
| group | string | true | none | none |
| start_date | string | true | none | none |
| elapsed_time | string | false | none | none |

| Name | Type | Required | Restrictions | Description |
|----------------------|--------|----------|--------------|-------------|
| priority | string | true | none | none |
| accept | string | true | none | none |
| queue | string | true | none | none |
| Fugaku:elapsed_limit | string | true | none | none |
| Fugaku:md | string | true | none | none |
| Fugaku:vnode | string | true | none | none |
| Fugaku:core | string | true | none | none |
| Fugaku:v_mem | string | true | none | none |
| Fugaku:v_pol | string | true | none | none |
| Fugaku:e_pol | string | true | none | none |
| Fugaku:rank | string | true | none | none |
| Fugaku:lst | string | true | none | none |
| Fugaku:ec | string | true | none | none |
| Fugaku:pc | string | true | none | none |
| Fugaku:sn | string | true | none | none |
| Fugaku:reason | string | true | none | none |

files_sync_post_request

```
{
  "path": "string",
  "direction": "to_online",
  "delete": false
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|------|--------|----------|--------------|--|
| path | string | true | none | Directory path to synchronized on API Server is given by <code>path</code> . |

| Name | Type | Required | Restrictions | Description |
|-----------|---------|----------|--------------|---|
| direction | string | false | none | <p>Direction of sync between API Sever Storage and Online Object Storage is given by <code>direction</code> (default is <code>from_online</code>). Bucket Name on Online Object Storage is predefined fixed value.</p> <p>* <code>to_online</code>: API Server Storage is source, and Online Object Storage is destination.</p> <p>* <code>from_online</code>: Online Object Storage is source, and API Server Storage is destination.</p> |
| delete | boolean | false | none | <p>If <code>true</code> is given by <code>delete</code> (default is <code>false</code>), files that exist in the destination but not in the source are deleted during sync.</p> |

Enumerated Values

| Property | Value |
|-----------|-------------|
| direction | to_online |
| direction | from_online |

files_sync_post_200_response

```
{
  "requestid": 0
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|-----------|---------|----------|--------------|------------------------------------|
| requestid | integer | true | none | Request ID of the synchronization. |

files_sync_requestid_get_200_response

```
{
  "result": "result",
  "status": "waiting"
}
```

Properties

| Name | Type | Required | Restrictions | Description |
|--------|--------|----------|--------------|--|
| status | string | true | none | Sync Status is set in <code>status</code> . If error, message is added after <code>error:</code> * <code>waiting</code> * <code>complete</code> * <code>error: parameter error</code> * <code>error: no such ID</code> |
| result | string | false | none | Status code is set in <code>result</code> if <code>status</code> is <code>complete</code> . If sync is successful, 0 is returned. If sync fails, non-zero value is returned. |

Enumerated Values

| Property | Value |
|----------|------------------------|
| status | waiting |
| status | complete |
| status | error: parameter error |
| status | error: no such ID |