

# Workflow Execution Service

# Table of Contents

1. Overview .....	1
1.1. Version information .....	1
1.2. URI scheme .....	1
1.3. Consumes .....	1
1.4. Produces .....	1
2. Executive Summary .....	2
3. Introduction .....	3
4. Standards .....	4
4.1. Authorization & Authentication .....	4
5. Paths .....	5
5.1. Run a workflow. This endpoint creates a new workflow run and returns the workflow ID to monitor its progress. The request may upload files that are required to execute the workflow identified as <code>workflow_attachment</code> . The parts supplied in <code>workflow_attachment</code> may include the primary workflow, tools imported by the workflow, other files referenced by the workflow, or files which are part of the input. The implementation should stage these files to a temporary directory and execute the workflow from there. These parts must have a Content-Disposition header with a "filename" provided for each part. Filenames may include subdirectories, but must not include references to parent directories with '..', implementations should guard against maliciously constructed filenames. The <code>workflow_url</code> is either an absolute URL to a workflow file that is accessible by the WES endpoint, or a relative URL corresponding to one of the files attached using <code>workflow_attachment</code> . The <code>workflow_params</code> JSON object specifies input parameters, such as input files. The exact format of the JSON object depends on the conventions of the workflow language being used. Input files should either be absolute URLs, or relative URLs corresponding to files uploaded using <code>workflow_attachment</code> . The WES endpoint must understand and be able to access URLs supplied in the input. This is implementation specific. See documentation for <code>WorkflowRequest</code> for detail about other fields.	6
5.1.1. Parameters .....	7
5.1.2. Responses .....	7
5.1.3. Consumes .....	7
5.1.4. Tags .....	7
5.2. List the workflow runs. This should be provided in a stable ordering, however the ordering of this list is implementation dependent. When paging through the list, the client should not make assumptions about live updates, but should assume the contents of the list reflect the workflow list at the moment that the first page is requested. To monitor a specific workflow run, use <code>GetRunStatus</code> or <code>GetRunLog</code> .	8
5.2.1. Parameters .....	8
5.2.2. Responses .....	8
5.2.3. Tags .....	9

5.3. Get detailed info about a workflow run.....	9
5.3.1. Parameters .....	9
5.3.2. Responses .....	9
5.3.3. Tags .....	9
5.4. Cancel a running workflow.....	9
5.4.1. Parameters .....	9
5.4.2. Responses .....	10
5.4.3. Tags .....	10
5.5. Get quick status info about a workflow run.....	10
5.5.1. Parameters .....	10
5.5.2. Responses .....	10
5.5.3. Tags .....	10
5.6. Get information about Workflow Execution Service. May include information related (but not limited to) the workflow descriptor formats, versions supported, the WES API versions supported, and information about general the service availability.	
5.6.1. Responses .....	11
5.6.2. Tags .....	11
6. Definitions .....	12
6.1. DefaultWorkflowEngineParameter .....	12
6.2. ErrorResponse.....	12
6.3. Log .....	12
6.4. RunId .....	13
6.5. RunListResponse.....	13
6.6. RunLog .....	13
6.7. RunRequest .....	13
6.8. RunStatus .....	14
6.9. ServiceInfo .....	14
6.10. State .....	15
6.11. WesObject.....	16
6.12. WorkflowTypeVersion .....	16

# Chapter 1. Overview

## 1.1. Version information

*Version* : 0.3.0

## 1.2. URI scheme

*BasePath* : /ga4gh/wes/v1

*Schemes* : HTTP, HTTPS

## 1.3. Consumes

- `application/json`

## 1.4. Produces

- `application/json`

# Chapter 2. Executive Summary

An executive summary summarising the major points of the document. To be added for [issue #37](#).

# Chapter 3. Introduction

Introduction setting the document's scope.

# Chapter 4. Standards

Standards incorporated (?)

## 4.1. Authorization & Authentication

Security issues around authentication and authorization need to be resolved and added into the specification. To be added for [issue #36](#).

# Chapter 5. Paths



**5.1. Run a workflow.** This endpoint creates a new workflow run and returns the workflow ID to monitor its progress. The request may upload files that are required to execute the workflow identified as `workflow_attachment`. The parts supplied in `workflow_attachment` may include the primary workflow, tools imported by the workflow, other files referenced by the workflow, or files which are part of the input. The implementation should stage these files to a temporary directory and execute the workflow from there. These parts must have a Content-Disposition header with a "filename" provided for each part. Filenames may include subdirectories, but must not include references to parent directories with '..', implementations should guard against maliciously constructed filenames. The `workflow_url` is either an absolute URL to a workflow file that is accessible by the WES endpoint, or a relative URL corresponding to one of the files attached using `workflow_attachment`. The `workflow_params` JSON object specifies input parameters, such as input files. The exact format of the JSON object depends on the conventions of the workflow language being used. Input files should either be absolute URLs, or relative URLs corresponding to files uploaded using `workflow_attachment`. The WES endpoint must understand and be able to access URLs supplied in the input. This is implementation specific. See documentation for WorkflowRequest for detail about other fields.

POST /runs

### 5.1.1. Parameters

Type	Name	Schema
FormData	<b>tags</b> <i>optional</i>	string (application/json)
FormData	<b>workflow_attachment</b> <i>optional</i>	< string (binary) > array
FormData	<b>workflow_engine_parameters</b> <i>optional</i>	string (application/json)
FormData	<b>workflow_params</b> <i>optional</i>	string (application/json)
FormData	<b>workflow_type</b> <i>optional</i>	string
FormData	<b>workflow_type_version</b> <i>optional</i>	string
FormData	<b>workflow_url</b> <i>optional</i>	string

### 5.1.2. Responses

HTTP Code	Description	Schema
200		<a href="#">RunId</a>
400	The request is malformed.	<a href="#">ErrorResponse</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.1.3. Consumes

- `multipart/form-data`

### 5.1.4. Tags

- WorkflowExecutionService

**5.2. List the workflow runs. This should be provided in a stable ordering, however the ordering of this list is implementation dependent. When paging through the list, the client should not make assumptions about live updates, but should assume the contents of the list reflect the workflow list at the moment that the first page is requested. To monitor a specific workflow run, use `GetRunStatus` or `GetRunLog`.**

GET /runs

### 5.2.1. Parameters

Type	Name	Description	Schema
Query	<b>page_size</b> <i>optional</i>	OPTIONAL The preferred number of workflow runs to return in a page. If not provided, the implementation should use a default page size. The implementation must not return more items than "page_size", but it may return fewer. Clients should not assume that if fewer than "page_size" items is returned that all items have been returned. The availability of additional pages is indicated by the value of "next_page_token" in the response.	integer (int64)
Query	<b>page_token</b> <i>optional</i>	OPTIONAL Token to use to indicate where to start getting results. If unspecified, return the first page of results.	string

### 5.2.2. Responses

HTTP Code	Description	Schema
200		<a href="#">RunListResponse</a>
400	The request is malformed.	<a href="#">ErrorResponse</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>

HTTP Code	Description	Schema
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.2.3. Tags

- WorkflowExecutionService

## 5.3. Get detailed info about a workflow run.

```
GET /runs/{run_id}
```

### 5.3.1. Parameters

Type	Name	Schema
Path	<b>run_id</b> <i>required</i>	string

### 5.3.2. Responses

HTTP Code	Description	Schema
200		<a href="#">RunLog</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>
404	The requested workflow run not found.	<a href="#">ErrorResponse</a>
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.3.3. Tags

- WorkflowExecutionService

## 5.4. Cancel a running workflow.

```
DELETE /runs/{run_id}
```

### 5.4.1. Parameters

Type	Name	Schema
Path	<b>run_id</b> <i>required</i>	string

### 5.4.2. Responses

HTTP Code	Description	Schema
200		<a href="#">RunId</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>
404	The requested workflow run wasn't found.	<a href="#">ErrorResponse</a>
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.4.3. Tags

- WorkflowExecutionService

## 5.5. Get quick status info about a workflow run.

```
GET /runs/{run_id}/status
```

### 5.5.1. Parameters

Type	Name	Schema
Path	<b>run_id</b> <i>required</i>	string

### 5.5.2. Responses

HTTP Code	Description	Schema
200		<a href="#">RunStatus</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>
404	The requested workflow run wasn't found.	<a href="#">ErrorResponse</a>
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.5.3. Tags

- WorkflowExecutionService

## 5.6. Get information about Workflow Execution Service. May include information related (but not limited to) the workflow descriptor formats, versions supported, the WES API versions supported, and information about general the service availability.

GET /service-info

### 5.6.1. Responses

HTTP Code	Description	Schema
200		<a href="#">ServiceInfo</a>
400	The request is malformed.	<a href="#">ErrorResponse</a>
401	The request is unauthorized.	<a href="#">ErrorResponse</a>
403	The requester is not authorized to perform this action.	<a href="#">ErrorResponse</a>
500	An unexpected error occurred.	<a href="#">ErrorResponse</a>

### 5.6.2. Tags

- WorkflowExecutionService

# Chapter 6. Definitions

## 6.1. DefaultWorkflowEngineParameter

A message that allows one to describe default parameters for a workflow engine.

Name	Description	Schema
<b>default_value</b> <i>optional</i>	The stringified version of the default parameter. e.g. "2.45".	string
<b>type</b> <i>optional</i>	Describes the type of the parameter, e.g. float.	string

## 6.2. ErrorResponse

An object that can optionally include information about the error.

Name	Description	Schema
<b>msg</b> <i>optional</i>	A detailed error message.	string
<b>status_code</b> <i>optional</i>	The integer representing the HTTP status code (e.g. 200, 404).	integer

## 6.3. Log

Name	Schema
<b>cmd</b> <i>optional</i>	< string > array
<b>end_time</b> <i>optional</i>	string
<b>exit_code</b> <i>optional</i>	integer (int32)
<b>name</b> <i>optional</i>	string
<b>start_time</b> <i>optional</i>	string
<b>stderr</b> <i>optional</i>	string
<b>stdout</b> <i>optional</i>	string

## 6.4. RunId

Name	Schema
<b>run_id</b> <i>optional</i>	string

## 6.5. RunListResponse

The service will return a RunListResponse when receiving a successful RunListRequest.

Name	Description	Schema
<b>next_page_token</b> <i>optional</i>	A token which may be supplied as "page_token" in workflow run list request to get the next page of results. An empty string indicates there are no more items to return.	string
<b>runs</b> <i>optional</i>	A list of workflow runs that the service has executed or is executing.	< <a href="#">RunStatus</a> > array

## 6.6. RunLog

Name	Description	Schema
<b>outputs</b> <i>optional</i>		<a href="#">WesObject</a>
<b>request</b> <i>optional</i>	The original request message used to initiate this execution.	<a href="#">RunRequest</a>
<b>run_id</b> <i>optional</i>		string
<b>run_log</b> <i>optional</i>		<a href="#">Log</a>
<b>state</b> <i>optional</i>		<a href="#">State</a>
<b>task_logs</b> <i>optional</i>		< <a href="#">Log</a> > array

## 6.7. RunRequest

To execute a workflow, send a run request including all the details needed to begin downloading and executing a given workflow.

Name	Description	Schema
<b>tags</b> <i>optional</i>		< string, string > map



Name	Description	Schema
<b>workflow_engine_parameters</b> <i>optional</i>	OPTIONAL Additional parameters can be sent to the workflow engine using this field. Default values for these parameters are provided at the ServiceInfo endpoint.	< string, string > map
<b>workflow_params</b> <i>optional</i>	REQUIRED The workflow run parameterization document (typically a JSON file), includes all parameterizations for the run including input and output file locations.	<a href="#">WesObject</a>
<b>workflow_type</b> <i>optional</i>		string
<b>workflow_type_version</b> <i>optional</i>		string
<b>workflow_url</b> <i>optional</i>	REQUIRED The workflow CWL or WDL document. When workflow attachments files are provided, the <b>workflow_url</b> may be a relative path corresponding to one of the attachments.	string

## 6.8. RunStatus

Small description of a workflow run, returned by server during listing

Name	Schema
<b>run_id</b> <i>required</i>	string
<b>state</b> <i>optional</i>	<a href="#">State</a>

## 6.9. ServiceInfo

A message containing useful information about the running service, including supported versions and default settings.

Name	Description	Schema
<b>auth_instructions_url</b> <i>optional</i>	A web page URL with information about how to get an authorization token necessary to use a specific endpoint.	string

Name	Description	Schema
<b>contact_info</b> <i>optional</i>	An email address or web page URL with contact information for the operator of a specific WES endpoint. Users of the endpoint should use this to report problems or security vulnerabilities.	string
<b>default_workflow_engine_parameters</b> <i>optional</i>	Each workflow engine can present additional parameters that can be sent to the workflow engine. This message will list the default values, and their types for each workflow engine.	< <a href="#">DefaultWorkflowEngineParameter</a> > array
<b>supported_file_system_protocols</b> <i>optional</i>	The filesystem protocols supported by this service, currently these may include common protocols such as 'http', 'https', 'sftp', 's3', 'gs', 'file', 'synapse', or others as supported by this service.	< string > array
<b>supported_wes_versions</b> <i>optional</i>		< string > array
<b>system_state_counts</b> <i>optional</i>	The system statistics, key is the statistic, value is the count of runs in that state. See the State enum for the possible keys.	< string, integer (int64) > map
<b>tags</b> <i>optional</i>		< string, string > map
<b>workflow_engine_versions</b> <i>optional</i>		< string, string > map
<b>workflow_type_versions</b> <i>optional</i>		< string, <a href="#">WorkflowTypeVersion</a> > map

## 6.10. State

- UNKNOWN: The state of the task is unknown.

This provides a safe default for messages where this field is missing, for example, so that a missing field does not accidentally imply that the state is QUEUED.

- QUEUED: The task is queued.

- INITIALIZING: The task has been assigned to a worker and is currently preparing to run.

For example, the worker may be turning on, downloading input files, etc.

- RUNNING: The task is running. Input files are downloaded and the first Executor has been started.

- PAUSED: The task is paused.

An implementation may have the ability to pause a task, but this is not required.

- COMPLETE: The task has completed running. Executors have exited without error

and output files have been successfully uploaded.

- EXECUTOR\_ERROR: The task encountered an error in one of the Executor processes. Generally, this means that an Executor exited with a non-zero exit code.
- SYSTEM\_ERROR: The task was stopped due to a system error, but not from an Executor, for example an upload failed due to network issues, the worker's ran out of disk space, etc.
- CANCELED: The task was canceled by the user.

*Type* : enum (UNKNOWN, QUEUED, INITIALIZING, RUNNING, PAUSED, COMPLETE, EXECUTOR\_ERROR, SYSTEM\_ERROR, CANCELED)

## 6.11. WesObject

An arbitrary structured object.

*Type* : object

## 6.12. WorkflowTypeVersion

Available workflow types supported by a given instance of the service.

Name	Description	Schema
<b>workflow_type_version</b> <i>optional</i>	an array of one or more acceptable types for the Workflow Type. For example, to send a base64 encoded WDL gzip, one could would offer "base64_wdl1.0_gzip". By setting this value, and the path of the main WDL to be executed in the workflow_url to "main.wdl" in the RunRequest.	< string > array