Select a definition   DataScape v1

DataScape 1.14.6 v1 OAS3

A REST Web API exposing Cae DataLife services and informations.

Apache 2.0

Servers

/Datascape

Authorize 🔓

Authorization ⌃

| POST | **/connect/token** | Permits to authenticate to DataScape and to get the access token, or to refresh it when it expires | ⌄ | 🔒 |
| POST | **/connect/logout** | Permits to logout from DataScape | ⌄ | 🔒 |

Actions ⌃

GET	**/v1/actions**	Returns the available actions.	⌄	🔒
POST	**/v1/actions**	Insert a new action.	⌄	🔒
GET	**/v1/actions/{actionId}**	Returns the details of an action.	⌄	🔒
PUT	**/v1/actions/{actionId}**	Modify an existing action.	⌄	🔒

DELETE	/v1/actions/{actionId}	Deletes an existing action.	⌄ 🔒

GET	/v1/action-priorities	Returns the localized name (description) of a priority given its id.	⌄ 🔓

POST	/v1/action-license-valid-test	Tests if a license is valid for a certain action.	⌄ 🔓

Alarms ⌃

GET	/v1/alarms/sentry-information		⌄ 🔓

GET	/v1/alarms/alarm-combination	Returns the available alarm combination.	⌄ 🔓

GET	/v1/alarms/entry-category	Returns the available entry categories.	⌄ 🔒

GET	/v1/alarms/entry-threshold-type	Returns the available entry threshold types.	⌄ 🔓

GET	/v1/alarms/entry-threshold-operator	Returns the available entry threshold operator.	⌄ 🔓

GET	/v1/alarms/invalid-data-behavior	Returns the available invalid data behavior.	⌄ 🔒

GET	/v1/alarms/alert-state	Returns the available alert state.	⌄ 🔒

GET	/v1/alarms/alert-state/{alertSetCentralId}/{alertSetId}	Returns the available alert state by alert set info	⌄ 🔓

GET	/v1/alarms/alert-state/{thresholdType}	Returns the threshold type's alert states.	⌄ 🔓

GET	`/v1/alarms`	Returns the available alarms by the alert set, if selected.	⌄ 🔓

DELETE	`/v1/alarms`	Deletes an existing alarm.	⌄ 🔓

POST	`/v1/alarms`	This method creates a new alarm with its params	⌄ 🔓

GET	`/v1/alarm-instances`	Returns the alarms than have created CAP notification, by the alert set, if selected, and by temporal range. If no temporal range is selected, the page will view the alarm from two day before now	⌄ 🔓

GET	`/v1/alarm-instances /{alarmInstanceId}`	Returns the alarm instance details.	⌄ 🔓

GET	`/v1/alarms /{alarmCentralId} /{alarmId}`	Returns the alarm details.	⌄ 🔓

PUT	`/v1/alarms /{alarmCentralId} /{alarmId}`	This method updates the alarm identified by id alarm with its params	⌄ 🔓

GET	`/v1/alarms/alert-sets`	Returns the available alert-sets.	⌄ 🔓

GET	`/v1/alarms/cap-category`	Returns the available cap categories.	⌄ 🔓

GET	`/v1/alarms/notification /sentryUrl`	Returns the Sentry URl.	⌄ 🔓

GET	`/v1/alarms/notification /dissemination-profiles`	Returns the list of profiles by Sentry.	⌄ 🔓

GET	`/v1/alarms/cap-language`	Returns the available cap languages.	⌄ 🔓

| GET | `/v1/alarms /notification-tags` | Returns the available tags that can be used in the notification messages. | ⌄ 🔓 |

AlertSets

| GET | `/v1/alarms/import- alerts-to-alarms` | This method changes the MapsME alarms in Overlook alarms | ⌃⌄ 🔓 |
| GET | `/v1/alertsets/priority` | Returns the available priority options. | ⌄ 🔓 |

| GET | `/v1/alertsets/station- alert-state- computation-mode` | Returns the available station state computation mode options. | ⌄ 🔓 |

| GET | `/v1/alertsets` | Returns the available alert sets. | ⌄ 🔓 |

| DELETE | `/v1/alertsets` | Deletes an existing alert set. | ⌄ 🔓 |

| POST | `/v1/alertsets` | This method creates a new alert set with its params | ⌄ 🔓 |

| PUT | `/v1/alertsets` | This method updates the alert set with its params | ⌄ 🔓 |

| GET | `/v1/alertsets/default- alertsets` | Returns the info of default alert sets. | ⌄ 🔓 |

| GET | `/v1/alertsets /{alertSetCentralId} /{alertSetId}` | Returns the info of alert set. | ⌄ 🔓 |

Basic Data ⌃

| GET | `/v1/basins` | Returns the list of the relevant basins for the current user | ⌄ 🔓 |

| GET | `/v1/all-basins` | Returns a list of basins | ⌄ 🔓 |

| GET | `/v1/rivers` | Returns a list of rivers | ⌄ 🔓 |

| GET | `/v1/alert-zones` | Returns a list of alert zones | ⌄ 🔓 |

| GET | `/v1/quantities/instruments-and-meas-units` | This method get the set of quantities with their instruments and meas-units | ⌄ 🔓 |

| GET | `/v1/instruments` | | ⌄ 🔓 |

| GET | `/v1/meas-units` | | ⌄ 🔓 |

| GET | `/v1/administrative-zones` | Returns the relevant administrative zones (countries, regions and provinces) for the current user | ⌄ 🔓 |

| GET | `/v1/quantities` | Returns the list of the relevant quantities for the current user (or all the quantities if requested) | ⌄ 🔓 |

| GET | `/v1/organizations` | Returns the list of the organizations available on UDB | ⌄ 🔓 |

Aegis Configuration ⌃

| GET | `/v1/aegis/configuration` | Returns the whole user configuration | ⌄ 🔓 |

| GET | `/v1/aegis/configuration/login` | Returns the configuration for the login page | ⌄ 🔓 |

| GET | `/v1/aegis/configuration/map` | Returns the configuration for the map pages | ⌄ 🔓 |

| GET | `/v1/aegis/configuration/alert-state-map` | Returns the configuration for the alert state map | ⌄ 🔓 |

| GET | `/v1/aegis/configuration/elements` | Returns the configuration for the elements page | ⌄ 🔓 |

| GET | `/v1/aegis/configuration/stations` | Returns the configuration for the stations page | ⌄ 🔓 |

PUT	`/v1/aegis/configuration/{section}/{param}`	Inserts or updates a single Aegis configuration parameter	⌄ 🔓

DELETE	`/v1/aegis/configuration/{section}/{param}`	Deletes a single Aegis configuration parameter	⌄ 🔓

POST	`/v1/aegis/configuration/area-elab/create`	Creates a new Area Elab	⌄ 🔓

PATCH	`/v1/aegis/configuration/area-elab/update-state`	Updates the state and the error message of an isopleth request	⌄ 🔓

GET	`/v1/aegis/configuration/area-elab/read/{isoplethRequestId}`	Returns the isopleth request	⌄ 🔓

GET	`/v1/aegis/configuration/area-elab/read-data/{isoplethRequestId}`	Returns the isopleth request data.	⌄ 🔓

GET	`/v1/aegis/configuration/area-elab/read`	Returns the isopleth requests by provided parameters. *** ATTENTION: only requests created in the last 30 minutes are considered.***	⌄ 🔓

DELETE	`/v1/aegis/configuration/area-elab/delete/{isoplethRequestId}`	Deletes an Area Elab (from Geoserver it is deleted, in the DB it's only marked with request state = DELETED)	⌄ 🔓

Clients ⌃

GET	`/v1/clients`	Returns a list of active clients	⌄ 🔓

GET	`/v1/jobs`	Returns the list of Jobs	⌄ 🔓

GET	`/v1/clients/recoveryElements`	Returns the list of stations and modules to forced rescue	⌄ 🔓

GET	`/v1/clients/postmen`	Returns the list of all postmen	⌄ 🔓

POST	`/v1/general-operation-start`	Tries to launch a general operation on a client	⌄ 🔓

POST	`/v1/elaboration-scheduler-operation-start`	Tries to launch an operation on Elaboration Scheduler	⌄ 🔓

POST	`/v1/scheduler-operation-start`	Tries to launch an operation on Scheduler	⌄ 🔓

POST	`/v1/dataRescue-operation-start`	Tries to launch an operation on DatRescue	⌄ 🔓

POST	`/v1/polling-operation-start`	Tries to launch an operation on Polling	⌄ 🔓

POST	`/v1/kernel-operation-start`	Tries to launch an operation on Scheduler	⌄ 🔓

GET	`/v1/general-operation-reply`	Returns the reply message by a client	⌄ 🔓

GET	`/v1/elaboration-scheduler-operation-reply`	Returns the reply message by Elaboration Scheduler	⌄ 🔓

GET	`/v1/scheduler-operation-reply`	Returns the reply message by Scheduler	⌄ 🔓

GET	`/v1/dataRescue-operation-reply`	Returns the reply message by DataRescue	⌄ 🔓

DELETE	`/v1/dataRescue-statistics-reply`	Returns the reply message by delete statistics request	⌄ 🔓

GET	`/v1/polling-operation-reply`	Returns the reply message by Polling	⌄ 🔓

Data

POST	`/v1/operation-stop`	Close the connection with clients	⌃ 🔓

GET	`/v1/data/{elementId}`	Returns the data for an element in a period of time.	⌄ 🔓

PUT	`/v1/data/{elementId}`	Writes measure data for an element.	⌄ 🔓

GET	`/v1/data-combo/{elementId}`	For internal use only. It returns a combination of element basic data, plausible values and extemporary values.	⌄ 🔓

GET	`/v1/data/last-time`	Returns the date and time of the last measure available for the user	⌄ 🔓

GET	`/v1/data/last-modified-days`	Returns the couples (date, elementId) of measure data that were inserted or changed after the passed threshold.	⌄ 🔓

GET	`/v3/data/last-time`	Returns the date and time of the last measure available for the user	⌄ 🔓

GET	`/v3/data-combo/{elementId}`	For internal use only. It returns a combination of element basic data, values and extemporary values.	⌄ 🔓

Data Exchange ⌃

GET	`/v1/data-exchange/transfer-merge-modes`	Returns the available merge mode for input and output transfers	⌄ 🔓

GET	`/v1/data-exchange/transfer-types`	Returns the available types for input and output transfers	⌄ 🔓

GET	`/v1/data-exchange/transfer-protocols`	Returns the available protocols for a transfer	⌄ 🔓

GET	`/v1/data-exchange /transfer-activations`	Returns the available activation modes for a transfer	⌄ 🔓
GET	`/v1/data-exchange /transfers`	Returns all the available local transfers (only those who have a local central id)	⌄ 🔓
POST	`/v1/data-exchange /transfers`	Creates a new local transfer	⌄ 🔓
PUT	`/v1/data-exchange /transfers`	Updates an existing transfer	⌄ 🔓
DELETE	`/v1/data-exchange /transfers /{transferCentralId} /{transferId}`	Deletes an existing transfer	⌄ 🔓
POST	`/v1/data-exchange/on- the-fly-transfer`	Tries to launch an on-the-fly transfer on Mover	⌄ 🔓
GET	`/v1/data-exchange/file- from-db`	Returns the content of the selected file	⌄ 🔓
GET	`/v1/data-exchange/on- the-fly-transfer-status`	Returns the state of an existing on-the-fly transfer	⌄ 🔓
GET	`/v1/data-exchange/paths`	Returns the file system paths used in the UDB-UDB transfers.	⌄ 🔓
POST	`/v1/data-exchange/paths`	Adds a new UDB-UDB file system path in configuration.	⌄ 🔓
PUT	`/v1/data-exchange/paths`	Updates an existing UDB-UDB file system path.	⌄ 🔓
DELETE	`/v1/data-exchange/paths /{pathName}`	Deletes an existing UDB-UDB file system path.	⌄ 🔓

| GET | **/v1/data-exchange /paths-statuses** | Returns for each transfer path a validity status flag, in order to check if the path is ready to be used by Mover or if something doesn't work. | ⌄ 🔓 |

| GET | **/v1/data-exchange /discover-central** | Executes the central discovery procedure in order to correctly configure and UDB-UDB transfer. | ⌄ 🔓 |

Elaboration Schedules ⌃

| GET | **/v1/elaboration- schedules** | It returns the list of all the elaboration schedules configured in UDB. | ⌄ 🔓 |

| PUT | **/v1/elaboration- schedules /{elaborationScheduleId }** | | ⌄ 🔓 |

Elements ⌃

| GET | **/v1/elements** | Returns the elements | ⌄ 🔒 |

| POST | **/v1/elements** | This method creates a new element with its params | ⌄ 🔒 |

| GET | **/v1/elements /{elementId}** | Returns the available informations for the selected element | ⌄ 🔓 |

| DELETE | **/v1/elements /{elementId}** | Deletes the selected element | ⌄ 🔓 |

| PUT | **/v1/elements /{elementId}** | This method update a new element with its params | ⌄ 🔓 |

| GET | **/v1/elements/element- series** | Returns a list of element series | ⌄ 🔓 |

| GET | **/v1/elements/types** | Returns a list type Ids | ⌄ 🔒 |

GET	`/v2/elements`	Returns the elements	⌄ 🔓

GET	`/v2/elements /{elementId}`	Returns the available informations for the selected element	⌄ 🔓

GET	`/v3/elements`	Returns the elements	⌄ 🔓

GET	`/v3/elements /{elementId}`	Returns the available informations for the selected element.	⌄ 🔓

Filters

GET	`/v1/filters /{filterType}`	Returns a list af available filters	⌄ 🔓

POST	`/v1/filters /{filterType}`	Creates a new prime filter using the input parameters as data	⌄ 🔓

GET	`/v1/filters /{filterCentralId} /{filterId}`	Returns The filter's info, identified by id and centralId	⌄ 🔓

DELETE	`/v1/filters /{filterCentralId} /{filterId}`	Deletes the filter by filterCentralId and filterId	⌄ 🔓

PUT	`/v1/filters /{filterCentralId} /{filterId}`	Modifies some parameters of a filter using its Id and central Id as keys	⌄ 🔓

GET	`/v1/filter-field- available-values /{filterType} /{filterFieldId}`	Returns The list of value, identified by filterFieldId and filterType	⌄ 🔓

GET	`/v1/filter-fields- operators`	Returns The list of operator of all fields	⌄ 🔓

GET	/v1/local-configurations	Returns the available local configurations	⌄ 🔓

ForeSide Configuration

GET	/v1/foreside /configuration	Returns the whole user configuration for ForeSide	⌄ 🔓

Formulas

GET	/v1/formulas	Return all the formulas	⌄ 🔓

POST	/v1/formulas	Create new formula	⌄ 🔓

PUT	/v1/formulas /{formulaOrgId} /{formulaId}	Modify a formula	⌄ 🔓

DELETE	/v1/formulas /{formulaOrgId} /{formulaId}	Delete a formula	⌄ 🔓

GET	/v1/formula-parse	Parse a formula and extract its parameters	⌄ 🔓

GET	/v1/formula-parse-with-check	Check if the number and names of the parameters of the new formula are correct, in case the formula is used by one or more elements	⌄ 🔓

HistoricalEvents

GET	/v1/historical-events /read/{elementId}	CRUD - READ METHOD: returns the list of historical events for the provided element id	⌄ 🔓

POST	/v1/historical-events /create	CRUD - CREATE METHOD: creates a new historical event	⌄ 🔓

DELETE	`/v1/historical-events/delete/{centralId}/{historicalEventId}`	CRUD - DELETE METHOD: deletes a historical event	⌄ 🔓
PATCH	`/v1/historical-events/update`	CRUD - UPDATE METHOD: updates *only the name* of a historical event	⌄ 🔓

Informations

GET	`/v1/info/version`		⌄ 🔓
GET	`/v1/info/udb-info`	Returns host name, database name and scenary name of the logged user	⌄ 🔓
GET	`/v1/info/broker-connection-test`		⌄ 🔒

Locations

GET	`/v1/locations`	Returns the locations of measurement (stations and remote sensors)	⌄ 🔓
GET	`/v3/locations`	Returns the locations of measurement (stations and remote sensors)	⌄ 🔓

Station and Element Custom Params

GET	`/v1/params`	Returns the list of params	⌄ 🔓
POST	`/v1/params`	Creates a new param	⌄ 🔓
GET	`/v1/param-types`	Returns the list of param types	⌄ 🔓
PUT	`/v1/params/{paramOrgId}/{paramId}`	Updates the param identified by paramOrgId and paramId	⌄ 🔒
DELETE	`/v1/params/{paramOrgId}/{paramId}`	Deletes the param identified by paramOrgId and paramId	⌄ 🔓

Registry ⌃

POST	`/v1/registry/upload`	This method does a copy of basic data files in a temporary directory, fixed for each session
GET	`/v1/registry/session`	This method creates a token for the session
GET	`/v1/registry`	This method imports the basic data to DataLife

Repeaters ⌃

GET	`/v1/repeaters`	Returns the list of repeaters, identified by routing

Sceneries ⌃

GET	`/v1/central-sceneries`	Return a list of local sceneries
POST	`/v1/sceneries`	Create a new scenery
PUT	`/v1/sceneries/{sceneryCentralId}/{sceneryId}`	Update the scenery identified by sceneryCentralId and sceneryId
DELETE	`/v1/sceneries/{sceneryCentralId}/{sceneryId}`	Delete the scenery identified by sceneryCentralId and sceneryId

Scenery ⌃

GET	`/v1/sceneries`	Returns all the available sceneries (for all centrals).

Selectives ⌃

GET	**/v1/selective-measures-request**	Asks DataLife to perform one or more selective measure(s).	⌄ 🔓
GET	**/v1/selective-measures-reply**	Returns the reply message from the Scheduler (A00F-06 Scheduler extemporaneous measure reply)	⌄ 🔓
GET	**/v1/selectives**	Returns the selective measures of the last 24h (one or more for each element: if more selectives for an element have been performed, they are all returned)	⌄ 🔓

Settings ⌃

GET	**/v1/settings/languages**	Returns the available UDB languages.	⌄ 🔓
GET	**/v1/settings/routing-types**	Returns the available routing types (i.e. all the different technologies available to reach a remote station).	⌄ 🔓
GET	**/v1/settings/general-settings**	Returns the main settings of this DataLife instance.	⌄ 🔓
PUT	**/v1/settings/general-settings/{name}**	Edits a main setting of this DataLife instance.	⌄ 🔒
POST	**/v1/settings/first-code-available/{codeRangeTypeId}**	Returns the first available id, during element or station creation	⌄ 🔓

SSOUsers ⌃

GET	**/v1/SSOUsers**	Returns the available SSO users.	⌄ 🔓
POST	**/v1/SSOUsers**	Creates a new SSO user	⌄ 🔓

GET `/v1/SSOUsers/{id}` Returns the SSO user, identidfied by id.

PUT `/v1/SSOUsers/{id}` Update the SSO user identified by id

DELETE `/v1/SSOUsers/{id}` Delete the SSO user by Id

StationAlarmReport

GET `/v1/station-alarm-report` Returns the all station alarm report (MHASTER and COMPACT).

GET `/v1/single-station-alarm-report` Returns the station alarm report detail

GET `/v1/station-alarm-report/first-photo` Returns the first site photo available for the station

GET `/v1/station-alarm-report/viewer-photo` Returns the full file path of the photo available for the station

GET `/v1/station-alarm-report/viewer-photo/{name}` Returns the full file path of the photo available for the station

Station configuration

GET `/v1/station-configurations/compact-configurations` Returns the compact configuration list. This method is deprecated and it will be used the GetAllCompactConfigurations method

GET `/v1/station-configurations/last-compact-configurations` Returns the complete compact configuration list (with stations without configuration)

GET	`/v1/station-configurations/all-compact-configurations/{stationCfgCentralId}/{stationId}`	Returns the complete compact configuration list (with stations without configuration) ⌄ 🔓
DELETE	`/v1/station-configurations`	Deletes the station configuration, identified by stationCentralId, stationId and cfgData ⌄ 🔓
GET	`/v1/station-configurations/compact-configurations/session`	This method creates a token for the session ⌄ 🔓
POST	`/v1/station-configurations/compact-configurations/upload`	This method does a copy of station configuration file in a temporary directory ⌄ 🔓
GET	`/v1/station-configurations/import-compact-configurations`	This method imports COmpact configuration in DataLife ⌄ 🔓
POST	`/v1/station-configurations/scheduler-cfg-download-request`	Tries to launch a download cfg request to Scheduler ⌄ 🔓
POST	`/v1/station-configurations/scheduler-cfg-upload-request`	Tries to launch an upload cfg request to Scheduler ⌄ 🔓
GET	`/v1/station-configurations/scheduler-cfg-download-reply`	Return the reply message later on download cfg request to Scheduler ⌄ 🔓
GET	`/v1/station-configurations/scheduler-cfg-upload-reply`	Return the reply message later on upload cfg request to Scheduler ⌄ 🔓

Stations ⌄

GET	`/v1/stations`	Returns the stations ⌄ 🔓

POST	`/v1/stations`	This method creates a new station with its params ⌄ 🔓

GET	`/v1/stations/{stationId}`	Returns the available informations for the selected station ⌄ 🔓

PUT	`/v1/stations/{stationId}`	This method updates a new station with its params ⌄ 🔓

DELETE	`/v1/stations/{stationId}`	⌄ 🔓

GET	`/v1/station-action-routings`	Returns the configured routings for each station that can be routed other than radio, and for each station with only radio routing but that has some obsolete no-radio routing configured. Each routing is per action. 🔓

PUT	`/v1/station-action-routings/{stationId}/{actionId}`	Update a routing for a given station and action. The chosen routing can be one of the available routing types or null in order to delete a former choice. 🔓

GET	`/v1/stations/station-series`	Returns a list of station series ⌄ 🔓

GET	`/v1/stations/countries`	Returns a list of countries ⌄ 🔓

GET	`/v1/stations/regions`	Returns a list of regions of a specific country ⌄ 🔓

GET	`/v1/stations/provinces`	Returns a list of provinces of a specific region ⌄ 🔓

GET `/v1/stations/cities` Returns a list of cities of a specific province ⌄ 🔓

GET `/v1/stations/basins` Returns a list of basins ⌄ 🔓

GET `/v1/stations/rivers` Returns a list of rivers ⌄ 🔓

GET `/v1/stations/alert-zones` Returns a list of alert zones ⌄ 🔓

GET `/v1/stations/routing-types` Returns a list of routing-types, with their protocols ⌄ 🔓

GET `/v1/stations/repeatersides` Returns a list of repeaters ⌄ 🔓

GET `/v1/stations/{stationId}/documents` Returns the names of the available documents for a station ⌄ 🔓

POST `/v1/stations/{stationId}/documents` Uploads a document for a station to the server ⌄ 🔓

DELETE `/v1/stations/{stationId}/documents/{name}` Deletes the specified document for a station ⌄ 🔓

GET `/v1/stations/{stationId}/documents/{name}` Returns (let the caller download) a document for a station by name No authentication required. ⌄ 🔓

GET `/v1/stations/{stationId}/photos` Returns the names of the available photos for a station (file names with extension) ⌄ 🔓

GET `/v1/stations/{stationId}/photos/{name}` Returns a photo for a station by name ⌄ 🔓

GET	/v1/stations /{stationId}/cam-photo	Returns a photo for a station by full path name	⌄ 🔓

GET	/v1/stations /{stationId}/first-cam- or-site-photo	Returns the first cam photo if available, otherwise the first site photo for the station	⌄ 🔓

GET	/v1/stations /{stationId}/first- photo	Returns the first site photo available for the station	⌄ 🔓

GET	/v1/stations /{stationId}/cam-photos /day	Returns the names of the last (in order of time) n photos for a station from video camera (file names with extension)	⌄ 🔓

GET	/v1/stations /{stationId}/cam-photos /last	Returns the names of the last (in order of time) n photos for a station from video camera (file names with extension)	⌄ 🔓

GET	/v1/stations /{stationId}/cam-photos /date/{datePart}	Returns the available years in the directory tree of the cam photos for a station	⌄ 🔓

GET	/v2/stations	Returns the stations; this is a corrected and improved implementation of v1/stations.	⌄ 🔓

GET	/v2/stations /{stationId}	Returns the available informations for the selected station	⌄ 🔓

GET	/v3/stations	Returns the stations; this is a corrected and improved implementation of v1/stations.	⌄ 🔓

StationStates

GET	/v3/stations /{stationId}	Returns the available informations for the selected station	◇ 🔓

GET	/v1/station-states	Returns the all station states (MHASTER and COMPACT).	⌄ 🔓

POST	/v1/change-station- state-request	Tries to change the station state	⌄ 🔓

GET	**/v1/change-station-state-reply**	Returns the reply message by Change station state request

Statistics ⌃

GET	**/v1/statistics /exclusions**	Returns the available information for the exclusions identified by from, to and type

POST	**/v1/statistics /exclusions**	Creates a new exclusion using the input parameters as data

GET	**/v1/statistics /exclusions-xlsx**	Returns the xlsx file with a list of exclusions, identified by from, to and type

GET	**/v1/statistics /acquisition-and-delay-raw-report**	Returns the raw statiscics data, group by station e/o element

GET	**/v1/statistics /acquisition-and-delay-raw-report-xlsx**	Returns the xslx file with raw statiscics data, group by station e/o element

PUT	**/v1/statistics /exclusions /{exclusionId}**	Modifies some parameters of an exclusion using its Id and central Id as keys

DELETE	**/v1/statistics /exclusions /{exclusionId}**	Deletes the exclusion with the Id given

GET	**/v1/statistics/station-availability/{year}**	Returns the availability values, group by station, for the specific year,

GET	**/v1/statistics/station-availability-xlsx /{year}**	Download file .xlsx with availability values, group by station, for the specific year,

GET	`/v1/statistics/station-availability/{year}/{stationId}`	Returns the availability values, group by station, for the specific year,	⌄ 🔒

GET	`/v1/statistics/latency/distribution`	Returns the latency time distribution of the year	⌄ 🔒

GET	`/v1/statistics/latency/distribution-xlsx`	Download a file .xlsx with latency time distribution of the year	⌄ 🔒

GET	`/v1/statistics/latency/above-threshold-ratio/{year}/{threshold}`	Returns the ratio between the number of polling requests, with latency greater than threshold, and the total requests, referring a specific year	⌄ 🔒

Strategies ⌃

GET	`/v1/strategies`	This methods returns a list of strategies	⌄ 🔓

POST	`/v1/strategies`	This method creates a new strategy with StrategyDTO parameters	⌄ 🔓

GET	`/v1/strategies/{strategyId}`	This method returns strategy's info, identified by id	⌄ 🔓

DELETE	`/v1/strategies/{strategyId}`	This method deletes the strategy identified by startegyId	⌄ 🔓

PUT	`/v1/strategies/{strategyId}`	This method modifies some parameters of a strategy, using its Id as key	⌄ 🔒

GET	`/v1/strategies/activities`	This method returns the list of strategy's action	⌄ 🔒

PUT	`/v1/strategies/{strategyId}/{active}`	This method modifies the active parameter, using strategy's Id as key	⌄ 🔒

Users ⌃

| GET | **/v1/users** | Returns the available users, for each user also the informations about linked actions and licenses are provided. | ⌄ 🔓 |

| POST | **/v1/users** | Creates a new user. | ⌄ 🔓 |

| GET | **/v1/users/{userId}** | Returns , for a specific user_id, the informations about linked actions and licenses are provided. | ⌄ 🔓 |

| PUT | **/v1/users/{userId}** | Updates an existing user. | ⌄ 🔓 |

| DELETE | **/v1/users/{userId}** | Deletes an existing user. The system user (i.e. the user with the lowest userId, usually 'CaeUser') cannot be deleted. | ⌄ 🔓 |

| GET | **/v1/users/licenses /{actionId}** | Returns the available users, for each user also the informations about linked actions and licenses are provided. | ⌄ 🔓 |

Utilities ⌃

| GET | **/v1/utilities/wms-relay /{remainder}** | | ⌄ 🔓 |

| GET | **/v1/utilities/wms- capabilities** | Returns the available WMS capabilities downloaded from the url passed. This method executes a GetCapabilities WMS standard request, and was created to bypass the CORS errors often present if this request was performed directly by a web browser. | ⌄ 🔓 |

VirtualElements ⌃

| GET | **/v1/calculated-elements** | Return all the calculated elements | ⌄ 🔓 |

POST	**/v1/calculated-elements**	Insert the new configuration of a virtual element	∨ 🔒

PUT	**/v1/calculated-elements**	Modify the configuration of a virtual element	∨ 🔒

GET	**/v1/calculated-elements-details**	Return the details of a calculated element	∨ 🔒

GET	**/v1/check-circular-references**	Check if there is a circularity reference	∨ 🔓

DELETE	**/v1/calculated-element/{calculatedElementId}**	Delete the configuration of a calculated element.	∨ 🔓

GET	**/v1/roughness** Return all the roughness	∨ 🔓

GET	**/v1/sections** Return all the sections	∨ 🔓

Schemas ∧

ActionLicenseDto

ActionOpenIddictApplicationDto

ActionParamWriteDto

ActionWriteDto

AlarmAlertSetDto

AlarmAlertStateDto

AlarmCombinationDto

AlarmDetailDto

AlarmDto

AlarmInstanceDto

AlarmNotificationDto

AlertSetDto

AlertStateDto

AreaElabDtoForCreation

AreaElabDtoForStateUpdate

CalculatedElementDetailDto

CapCategoryDto

CapLanguageDto

Category

CfgStatus

CommunicationActivityClusterDTO

CommunicationScheduleDTO

CompactConfiguration

CompactConfigurationDto

CompletionActivityLaunch

ConditionDto

ConditionValueDto

ConstantDto

DataCfgDto

DataElab

DataMeasType

DataPart

DataPartWrite

DataRescueOperation

DataRescueRequestDto

DataTiming

Date20

ElaborationActivityDto

ElaborationActivityPrecedingActivityDto

ElaborationScheduleDto

ElaborationScheduleTriggerDto

ElaborationSchedulerOperation

ElaborationSchedulerRequestDto

ElementAlertRowDto

ElementCustomParamDto

ElementCustomParamDtoV3

ElementCustomTrendDto

ElementDescrDto

ElementDetailDto

ElementDetailDtoV3

ElementDtoV2

ElementEntryCfgDto

ElementEntryStateDto

ElementField

ElementFieldV2

ElementFieldV3

ElementWriteDto

ElementsCfgDto

EnablingState

EntryDto

EntryGroupDto

EntryPriorityDto

EntryThresholdTypeDto

ExclusionDto

FilterOrigin

FilterType

FiltersCfgDto

FormulaDto

GapsDateRescueRequestDto

GeneralCfgDto

GeneralControlPanelOperation

HistoricalEventDto

InputDto

IntervalDto

InvalidDataBehaviorDto

JobType

KernelOperation

LastModifiedDayDto

MapCfgDto

MeasureWriteStatus

MsgA005_ReturnCode

MsgA00D_CommandCode

MsgA00D_PollingCommand

MsgA00D_PollingExecutionAction

MsgA00D_SubnetRequest

NewFormulaDto

OnTheFlyTransferRequestDto

OverrunBehavior

ParamSubclass

ParamWriteDto

ParameterDto

PathDto

PlausibleExtempDto

PlausibleExtempDtoV3

PollingOperation

PollingRequestDto

PrimeFilterDto

Priority

Routing

RoutingDto

RoutingDto2

RoutingWriteDto

SSOUserWriteDto

SceneryType

SceneryWritingDto

SchedulerOperation

SchedulerRequestDto

Section

SectionCalibrationParam

SectionPoint

StationAlertStateComputationModeDto

StationCustomParamDto

StationDetailDto

StationDetailDtoV3

StationDtoV2

StationField

StationFieldV2

StationFieldV3

StationStateRequestDto

StationWriteDto

StationsCfgDto

StrategyDTO

SynchroMode

TotalAssembledCfgDto

TransferDirection

TransferWriteDto

TypeFilter

TypeGroup

UserActionWriteDto

UserRoleParameterDto

UserRoleParameterType

UserWriteDto