Locust Forecast Web API User Guide

28 OCT 2021

The Locust forecast web API provides web accessible endpoints for automated programs to run and monitor a batch run. This document is for the web application version 0.4.4.

Access to the API requires a key issued at the time of registration. A key is unique to each user and it is used for authentication. There is a daily limit on the number of API calls each user can make. The limit is 250 calls per day (Eastern Time) and it may be changed without a prior notice to avoid system overload.

The base URL for a Locust forecast Web API is https://locusts.arl.noaa.gov. The base URL is to be prepended to an endpoint described below. If the endpoint you will use is /rest/v1/batch, the full URL for the endpoint is https://locusts.arl.noaa.gov/rest/v1/batch.

1. Submit a batch run

Endpoint	POST /rest/v1/batch	
Authentication	required	Available to registered users.
Request format	json or xml	Use the "Content-Type" header with application/json or application/xml.
Request body	required	This is a list of single swarm run inputs. A batch can contain up to 50 runs. Even when the batch has one swarm run, a list must be used.
Response format	json or xml	Use the "Accept" header with application/json or application/xml.
Response body		Returns the batch ID value that can be used for checking the run status. The number of API calls and the daily limit are also included in the response. An example in JSON: `{"batchId":7090,"dailyCounts":{"limit":250,"calls":1}}`.

Input fields for each single swarm run are shown below:

Name	Data Type	Description
name	string	(Required) Name of the single swarm run. Must consist of alphanumeric characters and/or underscores. The character length must be between 1 and 20.
latitude	number	(Required) Latitude of the starting location. Must be between -90 and 90 degrees.
longitude	number	(Required) Longitude of the starting location. Must be between -180 and 180 degrees.
height	number	(Required) Starting height in meters. Must be between 0 and 10,000.

height2	number	(Optional) Additional height of the starting location in meters. Must be between 0 and 10,000.
height3	number	(Optional) Additional height of the starting location in meters. Must be between 0 and 10,000.
meteorologicalData	string	(Required) Name of the meteorological data set. Must be GFS (which has one-degree spatial resolution) or GFS0p25 (for ¼-degree spatial resolution with shorter temporal range).
startDate	date	(Required) Start year, month, and day. Must use the YYYY-MM-DD format.
firstDayStartHour	integer	(Optional) Must be between 0 and 23 if used.
firstDayStartMinute	integer	(Optional) Must be between 0 and 59 if used.
firstDayEndingHour	integer	(Optional) Must be between 0 and 23 if used.
firstDayEndingMinute	integer	(Optional) Must be between 0 and 59 if used.
durationOfSimulation	integer	(Required) Number of days. Must be between 1 and 15. Note that the duration may be additionally limited by the meteorological data in use.
simulationDirection	integer	(Required) 0: forward in time, 1: backward in time.
nonstopFlight	boolean	(Required) If set to true, overnight stops are disabled.
takeoffTimeAfterSunrise	number	(Required) Takeoff time in hours after sunrise. For example, 2.5 means two and half hours later. Sunrise time is computed each day using the starting location. Must be between 0 and 4.0.
landingTimeBeforeSunset	number	(Required) Landing time in hours before sunset. For example, 2.5 means two and half hours prior to sunset. Sunset time is computed each day using the starting location. Must be between 0 and 4.0.
verticalMotion	integer	(Required) Must be between 0 and 4. For details, see Hysplit documentation for the vertical motion.
mapBackground	string	(Required) Map background. Supported values are terrain, toner, and arlmap.
spatialPlotRadius	number	(Required) Spatial radius of plots in km from the starting location. Must be 100.0 and 5000.0.
gisFileByDay	boolean	(Required) Create shapefiles by day.
gisFileByHeight	boolean	(Required) Create shapefiles by height.
gisFileByDayHeight	boolean	(Required) Create shapefiles by height and then by each day.
gisFileAllTrajectoriesInOne	boolean	(Required) Create shapefiles containing all trajectories.
useLineShapefile	boolean	(Optional) Use lines instead of points when creating shapefiles.Points will be used if not specified.
colorOpacity	number	(Required) Opacity (in %) of trajectories on plots. Must be between 0 and 100.

includeHysplitFile	boolean	(Optional) Include Hysplit files when creating the redistributable zip file. SETUP, CONTROL, Trajectory dump files, and others are in this category. It will be set to true if unspecified.
includeImage	boolean	(Optional) Include image files in the redistributable zip file. It will be set to true if unspecified.
includePostscript	boolean	(Optional) Include Postscript files in the redistributable zip file. It will be set to true if unspecified.
includePDF	boolean	(Optional) Include PDF files in the redistributable zip file. It will be set to true if unspecified.
includeShapefile	boolean	(Optional) Include GIS shapefiles in the redistributable zip file. It will be set to true if unspecified.
includeKMZ	boolean	(Optional) Include Google Earth file in the redistributable zip file. It will be set to true if unspecified.

A sample request body file in JSON containing two single swarm runs:

```
[{
  "name" : "swarm1",
  "latitude" : 4.0,
  "longitude" : 36.0,
  "height" : 500.0,
  "height2" : 1000.0,
  "meteorologicalData" : "GFS",
  "startDate" : "2020-04-20",
  "firstDayStartHour" : null,
  "firstDayStartMinute" : null,
  "firstDayEndingHour" : null,
  "firstDayEndingMinute" : null,
  "durationOfSimulation" : 3,
  "simulationDirection" : 0,
  "nonstopFlight" : false,
  "takeoffTimeAfterSunrise" : 2.0,
  "landingTimeBeforeSunset" : 1.0,
  "verticalMotion" : 4,
  "mapBackground" : "terrain",
  "spatialPlotRadius" : 500.0,
  "gisFileByDay" : true,
  "gisFileByHeight" : false,
  "gisFileByDayHeight" : false,
  "gisFileAllTrajectoriesInOne" : true,
  "useLineShapefile" : true,
  "colorOpacity" : 100,
  "includeHysplitFile" : true,
  "includeImage" : true,
```

```
      "includePostscript" : true,
      "includePDF" : true,
      "includeShapefile" : true,
      "includeKMZ" : true
}, {
      "name" : "swarm2",
      "latitude" : 4.5,
      "longitude" : 33.0,
      "height" : 500.0,
      "height2" : 1000.0,
      "meteorologicalData" : "GFS",
      "startDate" : "2020-04-20",
      "firstDayStartHour" : 8,
      "firstDayStartMinute" : 15,
      "firstDayEndingHour" : 17,
      "firstDayEndingMinute" : 30,
      "durationOfSimulation" : 3,
      "simulationDirection" : 0,
      "nonstopFlight" : false,
      "takeoffTimeAfterSunrise" : 2.0,
      "landingTimeBeforeSunset" : 1.0,
      "verticalMotion" : 4,
      "mapBackground" : "terrain",
      "spatialPlotRadius" : 500.0,
      "gisFileByDay" : true,
      "gisFileByHeight" : false,
      "gisFileByDayHeight" : false,
      "gisFileAllTrajectoriesInOne" : true,
      "useLineShapefile" : true,
      "colorOpacity" : 100,
      "includeHysplitFile" : true,
      "includeImage" : true,
      "includePostscript" : true,
      "includePDF" : true,
      "includeShapefile" : true,
      "includeKMZ" : true
}]
```

2. Checking batch run status

Endpoint	GET /rest/v1/batch/{**BATCH_ID**}	
Authentication	required	Available to registered users.

Request		Replace {BATCH_ID} with an actual batch ID.
Response format	json or xml	Use the "Accept" header with application/json or application/xml.
Response body		Returns the statuses of the batch run and all of its individual runs. The statue may be QUEUED, RUNNING, CRASHED, GRAPHICS_RUNNING, GRAPHICS_FAILED, COMPLETED, and EXPIRED. The number of API calls and the daily limit are also included in the response.

An example output after submitting a batch containing one single swarm run. The run is in the RUNNING state.

```
{
  "batchId":7090,
  "status":"RUNNING",
  "dailyCounts":{
    "limit":250,
    "calls":2
  },
  "runs":[
    {
      "id":7091,
      "name":"swarm1",
      "status":"RUNNING"
    }
  ]
}
```

Another example output after the run is completed. The state is COMPLETED.

```
{
  "batchId":7090,
  "status":"COMPLETED",
  "dailyCounts":{
    "limit":250,
    "calls":3
  },
  "runs":[
    {
      "id":7091,
      "name":"swarm1",
      "status":"COMPLETED"
    }
  ]
}
```