

User Manual for CreateRiverFloodScenario.py

Mark Armstrong, Field Support and Infrastructure Team, Office of Central Processing

1. Introduction

The CreateRiverFloodScenario.py script was created to facilitate the creation of test cases for RiverPro and Hazard Services. It was designed so that a user could just provide the location Id (LID) for a point and a scenario template, and the script would determine the actual data. It will also submit the resultant data files to EDEX for decoding and storage in the IHFS database. The user would not have to look up all of the reference data (primary physical element, flood stages, advisory stage, etc.) or manually enter data in a SHEF file or the hydro time series application.

Note: Running this script will add data to the database. If this is run for an AHPS point on an operational system, the data will be posted on AHPS if the LID/PE/TS is defined in hgstation. Use with caution.

2. Configuring the script

It is not necessary to configure the script, but it may be convenient to do so. There is a “Default parameter block” that defines the default settings for the command line arguments. It may be more convenient to edit this block rather than to enter all of the command line arguments each time the script is run, but it is not necessary.

3. Command Line Options

- l : LID (use a test ID). This argument is required.
- p : product ID and name of the resultant shef file (default=TEST)
- oi : obs Interval in minutes (default=15)
- fi : forecast interval in hours (default=6)
- ots : obs shef type source (default=RG)
- fts : forecast shef type source (default=FF)
- obsFile : file that contains the observed series template. This argument is required if --noObs or --fullFile are not specified.
- fcstFile : file that contains the forecast series template. This argument is required if --noFcst or --fullFile are not specified.
- fullFile : file that contains a template for a complete scenario with observed and forecast data. This argument requires that --startTime be specified.
- startTime : the reference time, or start time, of a full scenario given in epoch time in seconds (date +%s). This argument is only needed if --fullFile is specified.
- noObs : Do not generate an obs SHEF message
- noFcst : Do not generate a forecast SHEF message

Sample script execution syntax: `./CreateRiverFloodScenario.py -l DAWM2 -p TESTTEST --oi=30 --fi=12 --obsFile=obsTmpFile_new.txt --fcstFile=fcstTmpFile_new.txt`

4. The observed and forecast template files

A number of observed, forecast and full template files will be contained in a tar file (scenarios.tar) that is included with CreateRiverFloodScenario.py in the SCP. The format of all three file types is a series of multiplier/reference level pairs separated by a pipe (|) delimiter. The multiplier may be any floating point number and the reference levels are given as a character code. The meaning of the character codes are as follows:

W=action (advisory) stage/flow (riverstat.wstg/action_flow)

F=Flood stage/flow depending on the primary physical element (riverstat.fs/fq)

I=Minor flood stage/flow depending on the primary physical element (floodcat.minor_stage/minor_flow)

O=Moderate flood stage/flow depending on the primary physical element (floodcat.moderate_stage/moderate_flow)

A=Major flood stage/flow depending on the primary physical element (floodcat.major_stage/major_flow)

R=Record stage/flow from the crest table

Example of an observed template file:

Example of an obs series falling below flood stage

.95 F|.98 F|1.02 F|1.03 F|1.08 F|1.10 F

Lines that begin with '#' are treated as comments. In the observed file the data goes backward in time. The first pair is the most recent, so the .95 F applies to the observed time nearest to the reference (system time of script execution) time. .98 F applies to the reference time minus the observed time interval defined in the command line or from the default value. The first data value will be calculated as .95 * flood stage.

Example of a forecast template file:

Example of a forecast series that starts below flood stage and begins to fall

.95 F|.90 F|.82 F|.75 F|.69 F|.62 F|.61 F

The forecast file is in the same format, with a series of multiplier/reference level pairs that are pipe delimited. In the forecast file, the valid times are advancing forward in time. The .95 F pair applies to the reference time, which is the nearest value that falls on the forecast interval. For example, if the script is run at 14Z, the first valid time/reference time will be 18Z if a standard six hour forecast interval is used. If the flood stage is 10 feet, then the first valid time/value would be 18Z today/9.5 feet. The next pair, .90 F, would apply to 00Z the next day and have a value of 9.0 feet. Note that the basis time of the forecast will be the time that the script is run.

Note that the template format should not end with a '|'. A "fullFile" template file may contain multiple lines for readability, but no lines should end with a '|', or an empty pair will be inserted.

5. Output

The script will produce a SHEF file (format XXXXX_PRODUCTID.txt where XXXXX is the location ID and PRODUCTID is product ID in the command line) in the local /tmp directory and copy it to the EDEX SHEF end point (/awips2/edex/data/share/hydroapps/shefdecode/input/) available on all lx workstations. Unless the -noObs or -noFcst arguments are used, the file will contain two SHEF messages, one for the obs and one for the forecast.

6. Best practices

- On an operational site, use test points cloned from existing RiverPro forecast points.
- Sites are easily cloned within Hydrobase. Refer to that user manual for details.
- Choose an easily distinguished product ID for the -p argument. If you need to delete the data for any reason, it is easy to do so using the product ID. The product ID in the database will be the combination of the site ID and the product ID given in the -p argument.
- Since the output file is copied directly to the SHEF EDEX endpoint, it is not necessary to modify pqact.conf or shef.xml. The file is processed by the shefdecoder regardless of its Site ID or WMO ID.

7. Using -fullFile to create a multi-day event

The -fullFile functionality was designed to create a continuous multi-day event. The file contains a template for the entire event. The -startTime argument must be set in conjunction with -fullFile. When the script is run with these arguments, the current time is compared to the start time to determine where the obs/forecast cutoff is in the file. Every entry before the cutoff time becomes an observed value, while the values beyond become forecast values. Note that not all values beyond the cutoff time become forecast values in the output file. Some are skipped depending on the observed and forecast intervals that are chosen. Ideally, the script is run once or twice per day over several days to match the standard RFC forecast cycle.