

# LISFLOOD-FP

- [LISFLOOD-FP](#)
  - [LISFLOOD-FP forcing](#)
- [LISFLOOD-FP Module Adapter - summary](#)
  - [Example configuration generalAdapterRun](#)
    - [Export activities](#)
    - [Execute activities](#)
    - [Import activities](#)
  - [Requirements](#)
  - [Status](#)

## LISFLOOD-FP

LISFLOOD-FP is a two-dimensional hydrodynamic model specifically designed to simulate floodplain inundation in a computationally efficient manner over complex topography. It is capable of simulating grids for dynamic flood events and can take advantage of new sources of terrain information from remote sensing techniques such as airborne laser altimetry and satellite interferometric radar.

The model predicts water depths in each grid cell at each time step, and hence can simulate the dynamic propagation of flood waves over fluvial, coastal and estuarine floodplains. It is a non-commercial, research code developed as part of an effort to improve our fundamental understanding of flood hydraulics, flood inundation prediction and flood risk assessment.

### LISFLOOD-FP forcing

The data used for the forcing of Lisflood-fp is river levels and storm surge levels. Currently, both data are point source data, but we are developing a grid source data import for the storm surge data. Forcing of Lisflood-fp is through formatted ascii files, described in the lisflood-fp manuals. Data is exported as NetCDF data sources, and a python preprocessor prepares the Lisflood-fp control files from this. Output of Lisflood-fp is in ArcGrid ascii format, and a python postprocessor creates a NetCDF file which is subsequently imported into FEWS.

## LISFLOOD-FP Module Adapter - summary

This section describes the Lisflood-fp model adapter. The pre-adapter and post-adapter are Python modules, so a Python engine should be installed and accessible on the machine. The Python modules rely on the NetCDF4 and the Numpy libraries, which should also be installed. Assumption is a classical folder structure for the Module, containing of a Input folder, an Output folder and a WorkDir folder.

The GeneralAdapter exports two NetCDF-files into the Input folder:

- RiverWaterLevels.nc containing the river water levels for the river points
- StormSurgeLevels.nc containing the storm surge levels

The model adapter starts the model in the WorkDir by starting a file called run\_master.bat, which executes three tasks:

- Pre-adapter: extracting Lisflood control files from the Netcdf files in the input folder
- Running the Lisflood model
- Post-adapter: writing the output Netcdf file to the output folder

The GeneralAdapter imports the resulting NetCDF-file InundationDepth.nc.

### Example configuration generalAdapterRun

#### Export activities

### start up activities

```
<exportActivities>
  <exportNetcdfActivity>
    <exportFile>riverwaterlevels.nc</exportFile>
    <timeSeriesSets>
      <timeSeriesSet>
        <moduleInstanceId>Import_Lisflood_Riverlevels</moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>H.observed</parameterId>
        <locationSetId>BangladeshRivers.locations</locationSetId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="day" multiplier="1"/>
        <relativeViewPeriod unit="day" start="-10" end="0"/>
        <readWriteMode>add originals</readWriteMode>
      </timeSeriesSet>
    </timeSeriesSets>
  </exportNetcdfActivity>
  <exportNetcdfActivity>
    <exportFile>StormSurgelevels.nc</exportFile>
    <timeSeriesSets>
      <timeSeriesSet>
        <moduleInstanceId>Import_Lisflood_PointStormSurge<
/moduleInstanceId>
        <valueType>scalar</valueType>
        <parameterId>H.observed</parameterId>
        <locationSetId>sandwip.locations</locationSetId>
        <timeSeriesType>external historical</timeSeriesType>
        <timeStep unit="minute" multiplier="10"/>
        <relativeViewPeriod unit="day" start="-10" end="0"/>
        <readWriteMode>add originals</readWriteMode>
      </timeSeriesSet>
    </timeSeriesSets>
  </exportNetcdfActivity>
</exportActivities>
```

## Execute activities

### start up activities

```
<executeActivities>
  <executeActivity>
    <description>Run LisFlood_FP</description>
    <command>
      <executable>%ROOT_DIR%/workDir/run_master.bat</executable>
    </command>
    <timeOut>90000000</timeOut>
    <ignoreDiagnostics>true</ignoreDiagnostics>
  </executeActivity>
</executeActivities>
```

## Import activities

## start up activities

```
<importActivities>
  <importNetcdfActivity>
    <importFile>%ROOT_DIR%/Output/inundationDepth.nc</importFile>
    <timeSeriesSets>
      <timeSeriesSet>
        <moduleInstanceId>Lisflood_1h_inundation_hc</moduleInstanceId>
        <valueType>grid</valueType>
        <parameterId>InundationDepth</parameterId>
        <locationId>Sandwip_LF</locationId>
        <timeSeriesType>simulated historical</timeSeriesType>
        <timeStep unit="day"/>
        <readWriteMode>add originals</readWriteMode>
      </timeSeriesSet>
    </timeSeriesSets>
  </importNetcdfActivity>
</importActivities>
```

## Requirements

Lisflood-fp version 4.6.6

Python 2.7

Numpy 1.8

NetCDF 1.1

## Status

Lisflood-fp model adapter is still under development