

HEC-RAS Model Adapter

Contents

- [Contents](#)
- [Introduction](#)
- [Operating Forecasting Model](#)
- [Operating HEC-RAS Model and FEWS Adapter](#)
- [Download](#)
- [Interface between FEWS and HEC-RAS](#)
- [ID Mapping](#)
- [Directory structure](#)
 - [Technical details about communication between HEC-RAS adapter and DELFT-FEWS system.](#)
- [Description of the HEC-RAS data files](#)
- [Configuring HEC-RAS adapter](#)
 - [Add global properties for hecras model and binaries](#)
- [Overriding gate, levee breach settings](#)
- [List of input and output variables which can be exchanged with the Delft-FEWS system and HEC-RAS adapter](#)
- [Beste Practise Considerations](#)
- [Running model from FEWS](#)

Introduction

Note: The Delft-FEWS HEC-RAS adapter is maintained by [RMA](#).

The conceptual solution for the interface between HEC-RAS and FEWS has been illustrated in Figure 1. Two modes of working are identified that each support a basic use case. The modes of working are:

- Operational forecasting mode
- Calibration mode

The technical implementations for both modes of working are quite different. For running HEC-RAS in operational forecasting mode from FEWS, a software interface will be required that directly controls the model runs.

Calibration is considered as an activity that should be carried out offline from the forecasting system. This means that no direct control from FEWS will be required but a user will need to be able to migrate model datasets (calibrated schematizations) from the HEC-RAS calibration environment to the forecasting environment.

Present documentation will describe the first mode of operation. For details about operating model in the calibration mode please check standard HEC-RAS documentation.

Operating Forecasting Model

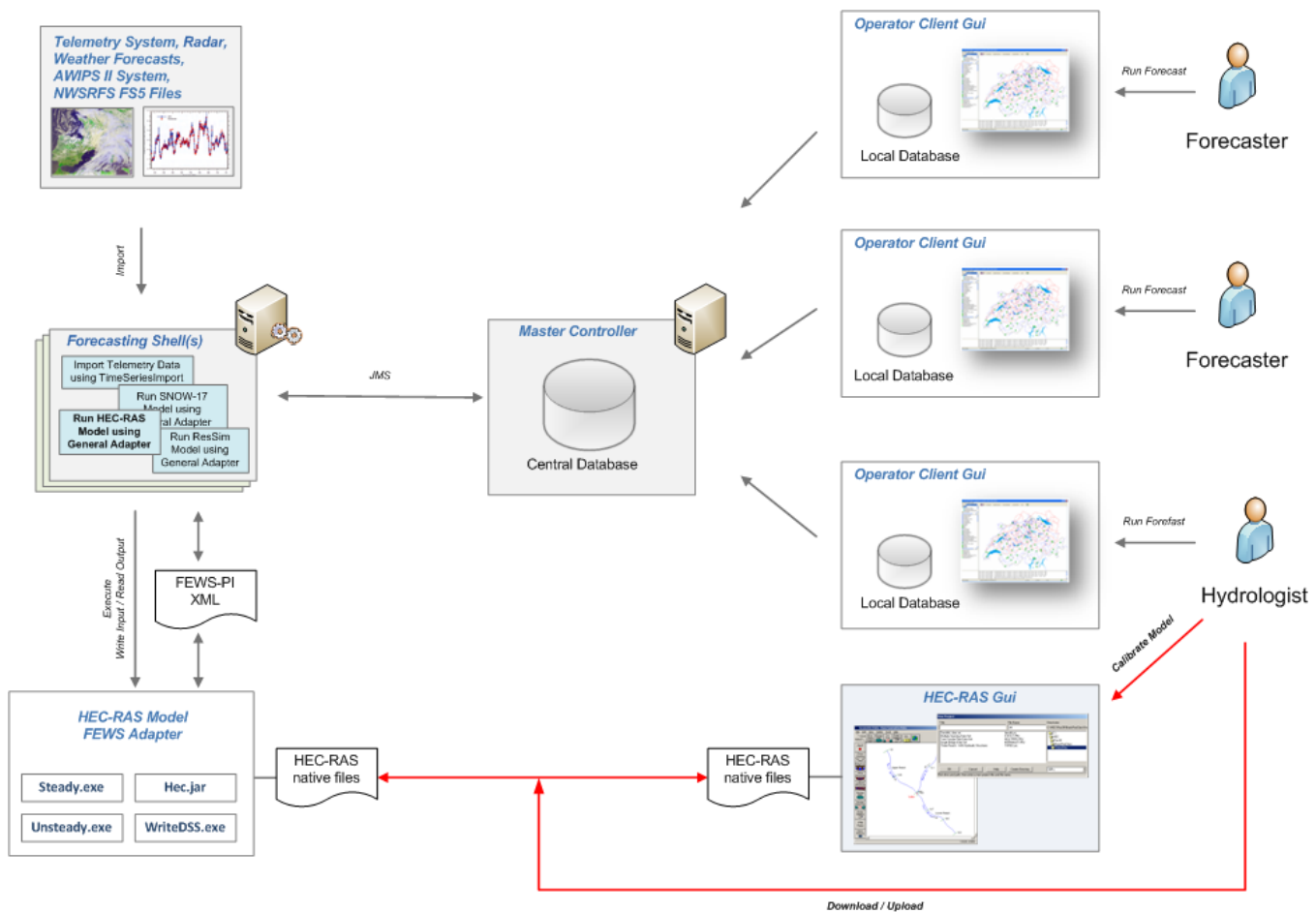


Figure 1 Components used to run forecasts using HEC-RAS model in the FEWS/CHPS system

Operating HEC-RAS Model and FEWS Adapter

The HEC-RAS model provides the compute engine for running a hydraulic model schematization for a section of a river or a part of a river system. The HEC-RAS Adapter forms the interface between the FEWS Forecasting Shell and the HEC-RAS model.

The HEC-RAS compute engine is, as its name suggests, the component that actually performs the HEC-RAS simulation. This simulation is controlled from the FEWS Adapter, and all run time data such as initial and boundary conditions, and parameter settings are passed through the adapter from and to the FEWS Forecasting Shell.

Download

Download of the model adapter is **not** available here: pls contact the adapter owner [RMA](#) or e-mail the [Delft-FEWS Product Management](#) for more information.

Configuration Manual for HEC-RAS versions 4.1: [Config manual RAS 4.1](#)

Configuration Manual for HEC-RAS versions 5.0.3: [Config manual RAS 5.0.3](#).

Configuration Manual for HEC-RAS versions 5.0.7: [Config manual RAS 5.0.7](#) and [RAS Mapper](#)

Configuration Manual for HEC-RAS version 6.0: [Config manual RAS 6.0](#)

Interface between FEWS and HEC-RAS

The FEWS Adapter for HEC-RAS forms the interface between the FEWS Forecasting Shell and the HEC-RAS model. The adapter accepts the request from the Forecasting Shell to run HEC-RAS, and imports the required data provided by the Forecasting Shell.

This data shall be provided in a standardized XML interface format, the FEWS Published Interface. Once a HEC-RAS run has been completed, relevant results are passed back to the Forecasting Shell in the form of the standardized XML interface format.

A schematic representation of the communication between the Forecasting Shell and the HEC-RAS model via the FEWS Adapter is shown in the diagram below.

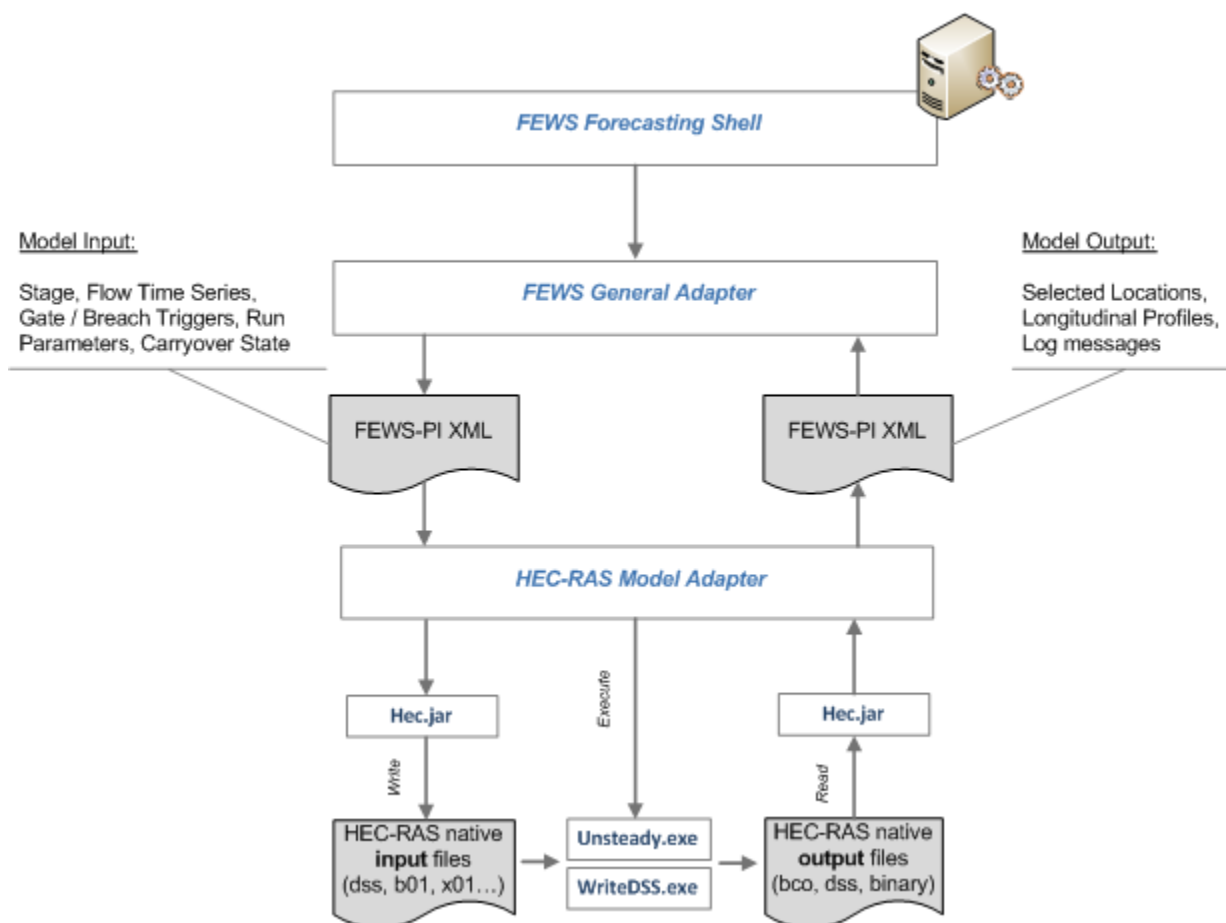


Figure 2 Data flows involved during run of HEC-RAS model FEWS adapter

The FEWS Adapter allows running of HEC-RAS by FEWS. The FEWS Adapter should be considered as a thin communication (software) layer on top the existing HEC-RAS engine. The adapter is tightly connected to the model engine. For longer term consistency, a FEWS adapter should therefore preferably be maintained by the owner of the model code, in this case HEC. The FEWS Adapter for HEC-RAS shall be developed by HEC or handed over to HEC upon completion.

The features of the are listed in the tables below.

Preprocessing 01	Clean up work and output folder
Preprocessing 02	Create module diagnostics file in Published Interface (PI) format
Preprocessing 03	Read the time series from PI time series
Preprocessing 04	Convert input PI time series into RAS *.b01 files
Launcher 01	Run HEC-RAS with run period and model alternative
Postprocessing 01	Open/create module diagnostics file in PI format
Postprocessing 02	Read the output time series from the RAS DSS and binary output files
Postprocessing 03	Write the time series to the output.xml
Postprocessing 04	Write the time series to the binary-output.xml
Postprocessing 05	Write the updated PI state file to export folder

ID Mapping

The location parameters used in FEWS can be coupled to HEC-RAS DSS path names through ID-mapping. The configuration files for ID-mapping should be created separately for each HEC-RAS model. Please consult [08 Mapping Id's flags and units](#) for more information on how to configure id mapping in FEWS system.

Directory structure

The data directories and configuration files that are required for operating the FEWS Adapter for HEC-RAS have been shown below.

Note that only binary and configuration files relevant to the HEC-RAS adapter are included, in a real configuration a lot more files can be involved used by another modules of the FEWS system.

```

+---bin
|           <FEWS binaries>
\---nerfc_sa

|
|   +---Config
|   |   +---ColdStateFiles
|   |   |       HECRAS_CONNECTICTUT_UpdateStates Default.zip....cold state files
|   |   |
|   |   +---IdMapFiles
|   |   |       IdExportHECRAS.xml..... custom mappings for the HEC-RAS variables and
locations
|   |   |
|   |   +---ModuleConfigFiles
|   |   |       HECRAS_CONNECTICTUT_Forecast.xml..... main configuration file of the adapter
|   |   |
|   |   +---ModuleDataSetFiles
|   |   |       HECRAS_CONNECTICTUT_UpdateStates.xml.....zipped hecras files, transported to Models
directory
|   |   |
|   |   \---ModuleParFiles
|   |   |       HECRAS_CONNECTICUT_Parameters..... configuration file which allows to override some
model and structure parameters
|   |
|   \---Models
|       \---hec/hecras
|           +---bin..... directory which contains all HEC-RAS executables
for Windows and Linux platforms
|           |       DSSWriter.exe..... generates binary file containing detailed model
output
|           |       dss_writer
|           |       GeomPreprocessor.exe..... converts geometry files from GUI ASCII format to
binary
|           |       geo_pre
|           |       Steady.exe..... performs steady flow simulations
|           |       steady
|           |       Unsteady.exe..... performs unsteady flow simulations
|           |       unsteady
|           |       DFORMD.DLL
|           |       DFORMDD.DLL
|           |       MSVCRTD.DLL
|           |       libmtsk.so.1
|           |       libwldelft_native.so
|           |       module-adapter-hec-ras.jar..... pre- and pos- adapter, Coverts HEC-RAS data files to
/from FEWS-PI format
|           |       nwsras.jar..... main library used by the adapter, reads and writes
HEC-RAS data files
|           |       heclib6-pc.dll
|           |       javaHeclib.dll
|           |       castor-0.9.5p.jar..... the rest of the files below are FEWS dependencies
used by adapter
|           |       commons-cli-1.1.jar
|           |       Delft_FEWS_castor.jar
|           |       Delft_FEWS_schemas.jar
|           |       Delft_PI.jar
|           |       Delft_PI_castor.jar
|           |       Delft_Util.jar
|           |       jaxp-api-1_3.jar
|           |       jaxp-dom-1_3.jar
|           |       jaxp-sax-1_3.jar
|           |       jaxp-xalan-1_3.jar
|           |       jaxp-xercesImpl-1_3.jar
|           |       log4j-1.2.14.jar
|           |       serializer.jar
|           |       TimeSeriesImport.jar

```

```

|      xalan.jar
|      xerces-c_2_8.dll
|      xercesImpl.jar
|      xml-apis.jar
|
|---connecticut
|      run_info.xml..... a file generated by FEWS containing paths, run
options
|
+---input..... input directory of the adapter, input FEWS-PI time
series files
|      inputs.xml
|
+---log..... log messages written by the hec-ras adapter
|      adapter.xml
|
+---output..... contains HEC-RAS output converted from the binary
and dss output files
|      output.xml
|
|---work..... working directory of the adapters
|      ctfld2ras.b01
|      ctfld2ras.b02
|      ctfld2ras.b03
|      ctfld2ras.c02
|      ctfld2ras.f04
|      ctfld2ras.g02
|      ctfld2ras.p01
|      ctfld2ras.p02
|      ctfld2ras.p05
|      ctfld2ras.prj
|      ctfld2ras.r01
|      ctfld2ras.r02
|      ctfld2ras.r03
|      ctfld2ras.r05
|      ctfld2ras.u01
|      ctfld2ras.u02
|      ctfld2ras.x02

```

Technical details about communication between HEC-RAS adapter and DELFT-FEWS system.

Communication between FEWS system and pre-/post- adapter strictly follows the FEWS Published Interface format.

Current implementation of the HEC-RAS adapter has all files required to run it (even in a stand-alone mode, without DELFT-FEWS system). The diagram below shows all dependencies from the FEWS libraries.



An adapter itself works only as a bridge between nwsras.jar library and DELFT-FEWS system. nwsras.jar provides a set of functions which allow to read /write all required HEC-RAS data files, including files used by the graphical user interface of HEC-RAS model.

For more technical details about functionality used by the adapter see [nwsras.doc](#) and [nwsras.testter.java](#) files in attachment.



Current version of HEC-RAS adapter is able to update all required HEC-RAS GUI files automatically when model is started from DELFT-FEWS. As result the user is able to get a complete model input generated by the DELFT-FEWS. This allows user to analyze model input in details using HEC-RAS GUI.

Description of the HEC-RAS data files

Table 1 List of files to be read and written by adapter

Extension	Description	pre-adapter input	pre-adapter output	post-adapter input	post-adapter output
.prj	project file	✓		✓	
.p01	plan files	✓		✓	
.g01	geometry files	✓		✓	
.f01	flow files				
.u01	unsteady flow files				
.b01	unsteady run files	✓	✓	✓	
.x01	input file for geometry preprocessor				
.r01	steady run file				
.O01	binary output file			✓	
.bco	model log file			✓	
.c01	output of geometry preprocessor				
.dss	input / output files			✓	
.hyd01	input file for geometry preprocessor				

Configuring HEC-RAS adapter

HEC-RAS model adapter follows standard way of integrating external models into the Delft-FEWS system by use of **General Adapter**. For more details about configuration of General Adapter please check [05 General Adapter Module](#).

A very important part of the configuration is defined under the **<exportRunFileActivity>** element. It contains path to the RAS project file, location of the RAS binary files and list of variables to be written into the output files. Additionally user may override logging level of the adapter to DEBUG in order to see more detailed output from the adapter. This is useful during configuration of the adapter since list of possible output variables that model can produce or list of input variables that can be consumed by the adapter are also printed to the log file.



List of output variables is defined under **outputTimeSeriesParametersFilter** and **outputLongitudinalProfileParametersFilter** items use [Regular Expressions](#). In most cases it is a list of variable names delimited with '|' character and for those variables where name can occur in another variable names (e.g. FLOW and FLOW AT GATE) it is necessary to use ^ as a prefix and \$\$ as a suffix of the variable. For example:

```
<string key="outputTimeSeriesParametersFilter" value="^STAGE$$|^FLOW$$|Hydr Radius R|Hydr Radius L|^Flow Area$$" />
```

Example configuration of the HEC-RAS adapter:

```
<?xml version="1.0" encoding="UTF-8"?>
<generalAdapterRun xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
schemaLocation="http://www.wldelft.nl/fews http://fews.wldelft.nl/schemas/version1.0/generalAdapterRun.xsd">
  <general>
    <description>hecras Model for Kennebec River</description>
    <rootDir>$HECRASMODELDIR$/kennebec</rootDir>
    <workDir>%ROOT_DIR%/work</workDir>
    <exportDir>%ROOT_DIR%/input</exportDir>
    <exportDataSetDir>%ROOT_DIR%/work</exportDataSetDir>
    <exportIdMap>IdExportHECRAS</exportIdMap>
    <exportUnitConversionsId>ExportHECRAS</exportUnitConversionsId>
    <importDir>%ROOT_DIR%/output</importDir>
    <importIdMap>IdImportHECRAS</importIdMap>
    <importUnitConversionsId>ImportHECRAS</importUnitConversionsId>
    <dumpFileDir>$GA_DUMPFILEDIR$/dumpFileDir</dumpFileDir>
    <dumpDir>%ROOT_DIR%</dumpDir>
    <diagnosticFile>%ROOT_DIR%/log/preadapter.xml</diagnosticFile>
  </general>
  <activities>
    <startUpActivities>
      <purgeActivity>
```

```

        <filter>%ROOT_DIR%/log/*.*/</filter>
    </purgeActivity>
    <purgeActivity>
        <filter>%ROOT_DIR%/input/*.*/</filter>
    </purgeActivity>
    <purgeActivity>
        <filter>%ROOT_DIR%/output/*.*/</filter>
    </purgeActivity>
    <purgeActivity>
        <filter>%ROOT_DIR%/work/*.*/</filter>
    </purgeActivity>
</setUpActivities>
<exportActivities>
    <exportStateActivity>
        <moduleInstanceId>HECRAS_KENNEBEC_UpdateStates</moduleInstanceId>
        <stateExportDir>%ROOT_DIR%/work</stateExportDir>
        <stateConfigFile>%ROOT_DIR%/work/state.xml</stateConfigFile>
        <stateLocations type="file">
            <stateLocation>
                <readLocation>Augusta.p02.rst</readLocation>
                <writeLocation>Augusta.p02.rst</writeLocation>
            </stateLocation>
        </stateLocations>
        <stateSelection>
            <warmState>
                <stateSearchPeriod unit="day" start="-10" end="-1"/>
            </warmState>
        </stateSelection>
    </exportStateActivity>
    <exportTimeSeriesActivity>
        <exportFile>%ROOT_DIR%/input/input.xml</exportFile>
        <timeSeriesSets>
            <timeSeriesSet>
                <moduleInstanceId>HECRAS_KENNEBEC_Preprocessing_UpdateStates<
/moduleInstanceId>

                <valueType>scalar</valueType>
                <parameterId>QINE</parameterId>
                <locationId>SIDM1ME</locationId>
                <timeSeriesType>simulated historical</timeSeriesType>
                <timeStep unit="hour" multiplier="1"/>
                <relativeViewPeriod unit="hour" end="0"/>
                <readWriteMode>read only</readWriteMode>
            </timeSeriesSet>
            <timeSeriesSet>
                <moduleInstanceId>HECRAS_KENNEBEC_Preprocessing_UpdateStates<
/moduleInstanceId>

                <valueType>scalar</valueType>
                <parameterId>STID</parameterId>
                <locationId>CASM1ME</locationId>
                <timeSeriesType>simulated historical</timeSeriesType>
                <timeStep unit="hour" multiplier="1"/>
                <relativeViewPeriod unit="hour" end="0"/>
                <readWriteMode>read only</readWriteMode>
            </timeSeriesSet>
        </timeSeriesSets>
    </exportTimeSeriesActivity>
    <exportDataSetActivity>
        <moduleInstanceId>HECRAS_KENNEBEC_UpdateStates</moduleInstanceId>
    </exportDataSetActivity>
    <exportParameterActivity>
        <fileName>params.xml</fileName>
        <moduleInstanceId>HECRAS_KENNEBEC_UpdateStates</moduleInstanceId>
    </exportParameterActivity>
    <exportRunFileActivity>
        <exportFile>%ROOT_DIR%/run_info.xml</exportFile>
        <properties>
            <string key="hecRasProjectFile" value="%ROOT_DIR%/work/Augusta.prj"/>
            <string key="hecRasBinDirectory" value="$HECRASBINDIR$"/>
            <string key="outputTimeSeriesParametersFilter" value="^STAGE$$|^FLOW$$"/>
            <string key="outputLongitudinalProfileParametersFilter" value="^STAGE$$|Hydr
Radius L"/>

```

```

        <string key="logLevel" value="DEBUG"/>
        <string key="skipBinaryOutput" value="false"/>
        <string key="hecRasEnvironment" value="
LD_LIBRARY_PATH=$HECRASBINDIR:$LD_LIBRARY_PATH"/>
    </properties>
    </exportRunFileActivity>
</exportActivities>
<executeActivities>
    <executeActivity>
        <command>
            <className>nl.wldelft.fews.adapter.hec.HecRasAdapter</className>
            <binDir>$HECRASBINDIR</binDir>
        </command>
        <arguments>
            <argument>%ROOT_DIR%/run_info.xml</argument>
        </arguments>
        <timeOut>1500000</timeOut>
    </executeActivity>
</executeActivities>
<importActivities>
    <importStateActivity>
        <stateConfigFile>%ROOT_DIR%/work/state.xml</stateConfigFile>
        <synchLevel>20</synchLevel>
    </importStateActivity>
    <importTimeSeriesActivity>
        <importFile>%ROOT_DIR%/output/output.xml</importFile>
        <timeSeriesSets>
            <timeSeriesSet>
                <moduleInstanceId>HECRAS_KENNEBEC_UpdateStates<
/moduleInstanceId>
                <valueType>scalar</valueType>
                <parameterId>SSTG</parameterId>
                <locationId>AUGM1ME</locationId>
                <timeSeriesType>simulated historical</timeSeriesType>
                <timeStep unit="hour" multiplier="1"/>
                <readWriteMode>add originals</readWriteMode>
            </timeSeriesSet>
            <timeSeriesSet>
                <moduleInstanceId>HECRAS_KENNEBEC_UpdateStates<
/moduleInstanceId>
                <valueType>scalar</valueType>
                <parameterId>SQIN</parameterId>
                <locationId>AUGM1ME</locationId>
                <timeSeriesType>simulated historical</timeSeriesType>
                <timeStep unit="hour" multiplier="1"/>
                <readWriteMode>add originals</readWriteMode>
            </timeSeriesSet>
        </timeSeriesSets>
    </importTimeSeriesActivity>
</importActivities>
</activities>
</generalAdapterRun>

```

The hecras files (b01.prj,u01.x01,run.sh etc.) are saved in the /Config/ModuleDataSet directory. These are copied to the /Model/hecras/<model>/work directory during the exportDataSet activity in the General Adapter file.

Add global properties for hecras model and binaries

The \$HECRASBINDIR\$ property is defined in the global.properties at the same level of the Config and Models directory:

```

HECRASMODELDIR=%REGION_HOME%/Models/hec/hecras
HECRASBINDIR=$HECRASMODELDIR$bin

```

Overriding gate, levee breach settings

In a current version of the HEC-RAS adapter the user may also override computational interval of the model as well as structure parameters.

This can be done using parameters file which need to be also referenced by the HEC-RAS module config file. See **<exportParameterActivity>** element in the general adapter configuration above.

An example belows shows list of the parameters which are supported for now.



Name of the structure defined in the HEC-RAS must exactly match group id and location of the structure (station, river, chainage) must be the same as locationId.

```
<parameters xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.wldelft.nl/fews/PI" xsi:
schemaLocation="http://www.wldelft.nl/fews/PI http://fews.wldelft.nl/schemas/version1.0/pi-schemas
/pi_modelparameters.xsd" version="1.5">
  <modifierType>HECRAS</modifierType>
  <group id="default" name="hec-ras run parameters">
    <parameter id="ComputationInterval">
      <description>Computation interval in minutes. Does not change interval of output data.</description>
      <intValue>5</intValue>
    </parameter>
  </group>

  <!-- Gate name and locationId should be equal to what is defined in the HEC-RAS gui -->
  <group id="Gate #1" name="hec-ras gate parameters">
    <locationId>CT River R1/18100</locationId>
    <!--

      Gate parameters depend on the mode selected in the RAS configuration files (gui),

      Posssible modes which can te set in the gui are:

          based on upstream WS (default)
          based on specified reference
          based on difference in stage

    -->

    <!-- parameters valid all modes -->
    <parameter id="RateOpen">
      <dblValue>0.05</dblValue>
    </parameter>

    <parameter id="RateClose">
      <dblValue>0.05</dblValue>
    </parameter>

    <parameter id="MaxOpen">
      <dblValue>20.0</dblValue>
    </parameter>

    <parameter id="MinOpen">
      <dblValue>0.0</dblValue>
    </parameter>

    <parameter id="InitOpen">
      <dblValue>3.0</dblValue>
    </parameter>

    <!-- parameters specific for "based on upstream WS" -->
    <parameter id="ZOpen">
      <description></description>
      <dblValue>4.0</dblValue>
    </parameter>

    <parameter id="ZClose">
      <description></description>
      <dblValue>3.0</dblValue>
    </parameter>

    <!-- parameters specific for "based on specified reference" -->
    <!--
    <parameter id="ReferenceWSType">
      <description>Valid values: Reach, RiverStation, StorageArea</description>
      <stringValue>Reach</stringValue>
    </parameter>
```

```

<parameter id="ReferenceWS">
  <description>Depending on the ReferenceWSType parameter</description>
  <stringValue>R1</stringValue>
</parameter>

<parameter id="referenceWSOpen">
  <description>Reference elevation at which gate begins to open</description>
  <dblValue>4.0</dblValue>
</parameter>

<parameter id="referenceWSClose">
  <description>Reference elevation at which gate begins to close</description>
  <dblValue>3.0</dblValue>
</parameter>
-->

<!-- parameters specific for "based on difference in stage" -->
<!--
<parameter id="stageDiffUSType">
  <description>Upstream Reach, RiverStation or StorageArea location for stage difference computation<
/description>
  <stringValue>Reach</stringValue>
</parameter>

<parameter id="stageDiffUS">
  <description>Depends on the stageDiffUSType parameter</description>
  <stringValue>Reach</stringValue>
</parameter>

<parameter id="stageDiffDSType">
  <description>Downstream River, Reach, RiverStation or StorageArea location for stage difference
computation</description>
  <stringValue>Reach</stringValue>
</parameter>

<parameter id="stageDiffDS">
  <description>Depends on the stageDiffDSType parameter</description>
  <stringValue>R1</stringValue>
</parameter>

<parameter id="stageDiffOpen">
  <description>Stage difference at which gate begins to open</description>
  <dblValue>0.1</dblValue>
</parameter>

<parameter id="stageDiffClose">
  <description>Stage difference at which gate begins to close</description>
  <dblValue>0.1</dblValue>
</parameter>
-->
</group>

<group id="Levee Breach" name="hec-ras levee breach parameters">
  <locationId>CT River R1/248658</locationId>

  <parameter id="IsActive">
    <description>true when breach is activated, otherwise model skips it during computations<
/description>
    <boolValue>>false</boolValue>
  </parameter>

  <parameter id="IsWSStart">
    <description>true if trigger for failure is WS elevation</description>
    <boolValue>true</boolValue>
  </parameter>

  <parameter id="ThresholdWS">
    <description>water surface elevation for breaching</description>
    <dblValue>3.4028E38</dblValue>
  </parameter>

```

```

<parameter id="ThresholdDuration">
  <description>threshold time (hours) for breaching</description>
  <dblValue>3.4028E38</dblValue>
</parameter>

<parameter id="StartDate">
  <description>Start date for breaching (e.g. 01MAR2001)</description>
  <stringValue></stringValue>
</parameter>

<parameter id="StartTime">
  <description>Start time for breaching (e.g. 1630)</description>
  <stringValue></stringValue>
</parameter>

<parameter id="CenterStation">
  <description>Center of breach (XS station / location)</description>
  <dblValue>8800.0</dblValue>
</parameter>

<parameter id="BottomWidth">
  <description>Final bottom width</description>
  <dblValue>500.0</dblValue>
</parameter>

<parameter id="BottomElevation">
  <description>Final bottom elevation</description>
  <dblValue>-10.0</dblValue>
</parameter>

<parameter id="LeftSideSlope">
  <description>Left side slope</description>
  <dblValue>2.0</dblValue>
</parameter>

<parameter id="RightSideSlope">
  <description>Right side slope</description>
  <dblValue>2.0</dblValue>
</parameter>

<parameter id="BreachTime">
  <description>Full formation time (hours)</description>
  <dblValue>1.0</dblValue>
</parameter>

<parameter id="WeirCoef">
  <description>Breach weir coefficient</description>
  <dblValue>2.6</dblValue>
</parameter>

<!-- parameter below are used only when IsPipe = true -->
<parameter id="IsPipe">
  <description>true if piping failure, false if overtopping</description>
  <boolValue>true</boolValue>
</parameter>

<parameter id="PipingCoefficient">
  <description>Piping coefficient (default is .8)</description>
  <dblValue>0.8</dblValue>
</parameter>

<parameter id="InitialPipingElevation">
  <description>Initial piping elevation</description>
  <dblValue>-0.5</dblValue>
</parameter>
</group>
</parameters>

```

List of input and output variables which can be exchanged with the Delft-FEWS system and HEC-RAS adapter

The HEC-RAS adapter is configured properly and forecast is performed from the Delft-FEWS system - a list of input and output variables will be written into the standard log file of the system. The location and variables are based on the active **<region>.b01** file of the HEC-RAS model configured in the GUI of HEC-RAS. Pre-adapter of the HEC-RAS will provide list of all possible input variables and locations in the following part of the log file:

```
...
156 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Found input at locations:
156 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl -      CT River, R1, 334752.0, Flow
Hydrograph
156 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl -      CT River, R1, 0.00, Stage
Hydrograph
...
```

Locations and variables listed after the line **Found input at locations:** can be configured in the Delft-FEWS as a part of adapter input, e.g. **input.xml** in this case may contain something like the lines below:

input.xml

```
<TimeSeries
  xsi:schemaLocation="http://www.wldelft.nl/fews/PI http://fews.wldelft.nl/schemas/version1.0/pi-schemas
/pi_timeseries.xsd"
  version="1.2" xmlns="http://www.wldelft.nl/fews/PI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <timeZone>0.0</timeZone>
  <series>
    <header>
      <type>instantaneous</type>
      <locationId>CT RIVER R1/334752.0</locationId>
      <parameterId>Flow Hydrograph</parameterId>
      <timeStep unit="second" multiplier="3600"/>
      <startDate date="2008-11-06" time="12:00:00"/>
      <endDate date="2008-11-08" time="12:00:00"/>
      <missVal>-999.0</missVal>
      <stationName>Connicut River at Thompsonville</stationName>
      <units>cms</units>
    </header>
    <event date="2008-11-06" time="12:00:00" value="14.98" flag="0"/>
    <event date="2008-11-06" time="13:00:00" value="14.705" flag="0"/>
    <event date="2008-11-06" time="14:00:00" value="14.43" flag="0"/>
    <event date="2008-11-06" time="15:00:00" value="14.155" flag="0"/>
    <event date="2008-11-06" time="16:00:00" value="13.88" flag="0"/>
    <event date="2008-11-06" time="17:00:00" value="13.605" flag="0"/>
    ...
  </series>
</TimeSeries>
```

Note that **<parameterId>** and **<locationId>** are exactly the same as the variables and locations listed in the log file.

In the same way list of all output variables and locations can be found in the post-adapter log output, for example:

```

...
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Profile, Profile number.,
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - W.S. Elev, Calculated water surface
from energy equation., (ft)
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - E.G. Elev, Energy gradeline for
given WSEL., (ft)
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Max Chl Dpth, Maximum main channel
depth., (ft)
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Min Ch El, Minimum Channel
Elevation., (ft)
84843 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Q Left, Flow in left overbank.,
(cfs)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Q Channel, Flow in main channel.,
(cfs)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Q Right, Flow in right overbank.,
(cfs)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Q Total, Total flow in cross
section., (cfs)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Flow Area, Total area of cross
section active flow., (sq ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Flow Area L, Area of left overbank
active flow., (sq ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Flow Area Ch, Area of main channel
active flow., (sq ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - Flow Area R, Area of right overbank
active flow., (sq ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - W.P. Total, Wetted perimeter of
total cross section., (ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - W.P. Left, Wetted perimeter of left
overbank., (ft)
84859 [main] INFO nl.wldelft.fews.adapter.hec.HecRasDataAccessFacadeImpl - W.P. Channel, Wetted perimeter of
main channel., (ft)
...

```

Note that parameter names to be written into the output FEWS-PI file will contain only the short name of the parameter, e.g.: **Q Right** and not **Flow in right overbank., (cfs)**.

The variables listed here will be written into the file specified as a "--output-binary-pi-file=<path>" argument of the post-adapter. Example of the resulting FEWS-PI xml can be found below:

Example of the FEWS-PI containing binary output of the HEC-RAS model

```
<TimeSeries
  xsi:schemaLocation="http://www.wldelft.nl/fews/PI http://fews.wldelft.nl/schemas/version1.0/pi-schemas
/pi_timeseries.xsd"
  version="1.2" xmlns="http://www.wldelft.nl/fews/PI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <timeZone>0.0</timeZone>
  <series>
    <header>
      <type>instantaneous</type>
      <locationId>CT River R1/334752.0</locationId>
      <parameterId>W.S. Elev</parameterId>
      <timeStep unit="second" multiplier="3600"/>
      <startDate date="2008-11-06" time="00:00:00"/>
      <endDate date="2008-11-08" time="00:00:00"/>
      <missVal>NaN</missVal>
      <units>[?]</units>
    </header>
    <event date="2008-11-06" time="00:00:00" value="32.06013" flag="0"/>
    <event date="2008-11-06" time="01:00:00" value="32.06013" flag="0"/>
    <event date="2008-11-06" time="02:00:00" value="32.034" flag="0"/>
    <event date="2008-11-06" time="03:00:00" value="32.03394" flag="0"/>
    ...
    <event date="2008-11-07" time="23:00:00" value="32.03618" flag="0"/>
    <event date="2008-11-08" time="00:00:00" value="32.03598" flag="0"/>
  </series>
  <series>
    <header>
      <type>instantaneous</type>
      <locationId>CT River R1/334752.0</locationId>
      <parameterId>E.G. Elev</parameterId>
      <timeStep unit="second" multiplier="3600"/>
      <startDate date="2008-11-06" time="00:00:00"/>
      <endDate date="2008-11-08" time="00:00:00"/>
      <missVal>NaN</missVal>
      <units>[?]</units>
    </header>
    <event date="2008-11-06" time="00:00:00" value="32.06734" flag="0"/>
    <event date="2008-11-06" time="01:00:00" value="32.06734" flag="0"/>
    <event date="2008-11-06" time="02:00:00" value="32.056885" flag="0"/>
    ...
  </series>
</TimeSeries>
```

Additionally to the variables available in the binary output of the HEC-RAS, usually called <file>.O01, a DSS output is available. In most cases it contains FLOW and STAGE variables. Example of the FEWS-PI generated from the DSS file is given below:

Example of the FEWS-PI containing DSS output of the HEC-RAS model

```
<TimeSeries
  xsi:schemaLocation="http://www.wldelft.nl/fews/PI http://fews.wldelft.nl/schemas/version1.0/pi-schemas
/pi_timeseries.xsd"
  version="1.2" xmlns="http://www.wldelft.nl/fews/PI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <timeZone>0.0</timeZone>
  <series>
    <header>
      <type>instantaneous</type>
      <locationId>CT RIVER R1/0.00</locationId>
      <parameterId>FLOW</parameterId>
      <timeStep unit="second" multiplier="3600"/>
      <startDate date="2008-11-06" time="00:00:00"/>
      <endDate date="2008-11-08" time="00:00:00"/>
      <missVal>NaN</missVal>
      <units>CFS</units>
    </header>
    <event date="2008-11-06" time="00:00:00" value="24.38823" flag="0"/>
    <event date="2008-11-06" time="01:00:00" value="-5.8442316" flag="0"/>
    <event date="2008-11-06" time="02:00:00" value="68.705124" flag="0"/>
    <event date="2008-11-06" time="03:00:00" value="391.09784" flag="0"/>
    ...
    <event date="2008-11-07" time="23:00:00" value="438.6425" flag="0"/>
    <event date="2008-11-08" time="00:00:00" value="-5259.6562" flag="0"/>
  </series>
  <series>
    <header>
      <type>instantaneous</type>
      <locationId>CT RIVER R1/0.00</locationId>
      <parameterId>STAGE</parameterId>
      <timeStep unit="second" multiplier="3600"/>
      <startDate date="2008-11-06" time="00:00:00"/>
      <endDate date="2008-11-08" time="00:00:00"/>
      <missVal>NaN</missVal>
      <units>FEET</units>
    </header>
    <event date="2008-11-06" time="00:00:00" value="5.0" flag="0"/>
    <event date="2008-11-06" time="01:00:00" value="5.0" flag="0"/>
    <event date="2008-11-06" time="02:00:00" value="5.0" flag="0"/>
    <event date="2008-11-06" time="03:00:00" value="5.0" flag="0"/>
    ...
  </series>
</TimeSeries>
```

Beste Practise Considerations

The HEC-RAS adapter requires quite some synergy between FEWS, Java and the actual HEC-RAS software versions. While a FEWS General Adapter config that uses embedded Java adapters is technically preferred above a custom JRE, it very sensitive to any software update. From past implementations we experience that using old models that run with older RAS versions, do not run anymore with actual adapter and Java versions. Therefore from an organizational and robustness perspective, it is preferred to have your both softwares as independent as possible, so with a custom JRE.

So instead of using the below default config:

Embedded GA execute activity, running with FEWS 2016.02 and HEC-RAS v5

```
<executeActivity>
  <command>
    <className>nl.wldelft.fews.adapter.hec.HecRasAdapter</className>
    <binDir>${HECRASBINDIR}</binDir>
  </command>
  <arguments>
    <argument>%ROOT_DIR%/run_info.xml</argument>
  </arguments>
  <timeOut>3600000</timeOut>
</executeActivity>
```

one can better use a config like:

Independent GA execute activity running with FEWS 2023.02 and HEC-RAS v5

```
<executeActivity>
  <command>
    <className>nl.wldelft.fews.adapter.hec.HecRasAdapter</className>
    <binDir>$HECRASBINDIR$</binDir>
    <!-- use old Java 8, while 2023.02 uses 11-->
    <customJreDir>$CUSTOMJRE_64B_DIR$</customJreDir>
    <!-- in case of custom JRE, pass correct default Parser definition-->
    <jvmArg>-Dorg.xml.sax.parser=com.sun.org.apache.xerces.internal.parsers.SAXParser</jvmArg>
  </command>
  <arguments>
    <argument>%ROOT_DIR%/run_info.xml</argument>
  </arguments>
  <console>
    <!-- dump console in txt file, so in case of a crash, you see what happend-->
    <redirectToLogFile>%ROOT_DIR%/log/adapter_console_logging.txt</redirectToLogFile>
  </console>
  <timeOut>3600000</timeOut>
</executeActivity>
```

Running model from FEWS

Check [Using Delft-FEWS - User Guide](#) on how to run configured model from the Delft-FEWS system.