XBeach Adapter

XBeach

XBeach is a two-dimensional model for wave propagation, long waves and mean flow, sediment transport and morphological changes of the nearshore area, beaches, dunes and backbarrier during storms.

XBeach Module Adapter - summary

This page describes the XBeach module adapter, its functions, and provides an example for configuring a XBeach run in FEWS.

The pre-adapter creates the model specific output by replacing tags in template files:

- zs0file.txt for water level input file zs0file.txt
- bcfile.txt and bc.timeXXX.sp2 for SWAN spectrum input files
- params.txt for run information

Furthermore, the pre-adapter writes log messages to a log file called xbeach.log.

There is no post-adapter, since XBeach can be configured to use netcdf as output format that is readable by FEWS.

XBeach pre-adapter

Model pre-adapter for running a XBeach model from Delft-FEWS.

Class name: nl.deltares.xbeach.XBeachPreAdapter

Properties

No specific properties need to be configured for a model run.

There is however extra functionality provided via the properties but this is not needed if the model is setup correctly.
Configured properties starting with "PARAM_" will be added or replaced literally without "PARAM_" (and in lower case) in the params.txt file which defines the parameters for an XBeach model run.
This functionality should only be used as a debug option and only by someone that understands the XBeach properties in params.txt. To setup a model make sure you use a correctly configured params.txt so this functionality is not needed.

Notes for users

- For all files that are written by this adapter, if the file to be written already exists, then it will be overwritten.
- This program writes log messages to a log file called xbeach.log.
- This program does not make use of a template file, parameters are automatically added and replaced in params.txt without the use of tags.
- This program uses the information in the specified netcdf run file as input and uses this information to do the following actions:
  1. Create the water level input file: zs0file.txt, see water level conversion
  2. Create the SWAN spectrum input files bcfile.txt and bc.timeXXX.sp2, see wave spectrum conversion
  3. Replace or add the parameters 'tstart' and 'tstop' in params.txt, see parameter conversion

System requirements

- This program needs Java version 1.7 or higher.
- This program needs the following Java libraries:
  - castor-0.9.5.jar
  - commons-httpclient-3.0.1.jar
  - Delft_Util.jar

- fews-xbeach-adapter.jar
- log4j-1.2.14.jar
- netcdf-4.2.jar
- slf4j-api-1.5.6.jar
- slf4j-log4j12-1.5.6.jar
- xercesImpl.jar

Xbeach post-adapter

There is no need for an Xbeach post-adapter since XBeach can be configured to use netcdf as output format that is readable by FEWS.

Example configuration generalAdapterRun

A complete example model run configuration file can be found here: XBeachAdapterRun.xml
Information how to prepare a FEWS environment to be able to use the FEWS model adapter can be found here: XBeach FEWS setup

Start up activities

As a first activity it can be useful to delete all files present in the workDir, if for example it would be filled with files from a previous run.

start up activities

```
<startUpActivities>
        <purgeActivity>
                <filter>workDir*</filter>
        </purgeActivity>
</startUpActivities>
```

Export activities

The first steps in the general adapter run are the data set, netcdf and run file export activities. The <exportDataSetActivity> will extract a zip file with the module instance id as file name located in "Config\ModuleDataSetFiles\" of the FEWS environment to the workdir. The <exportNetcdfActivity>'s will be a netcdf file (bcfile.nc) containing Swan wave spectra over time and a netcdf file (zs0file.nc) containing water level over time. The <exportNetcdfRunFileActivity> will be a netcdf run file that contains information needed by the pre-adapter. The information will be automatically filled by the general adapter but properties can be configured as extra information. For example properties starting with "PARAM_" will be added or replaced literally without "PARAM_" (and in lower case) in the params.txt file which defines the parameters for an XBeach model run. An example is given in the config below as <string key="PARAM_OUTPUTFORMAT" value="netcdf"/> this adds or replaces parameter 'outputformat' in params.txt and assigns the value 'netcdf'. All parameters should however be already correctly set in params.txt so these property should not be necessary.

export activities

```
                <exportActivities>
                        <exportDataSetActivity>
                                <moduleInstanceId>Run_XBeach</moduleInstanceId>
                        </exportDataSetActivity>
                        <exportNetcdfActivity>
                                <exportFile>bcfile.nc</exportFile>
                                <timeSeriesSets>
                                        <timeSeriesSet>
                                                <moduleInstanceId>Run_XBeach</moduleInstanceId>
                                                <valueType>scalar</valueType>
                                                <parameterId>EnDens</parameterId>
                                                <domainParameterId>AFREQ</domainParameterId>
                                                <domainParameterId>NDIR</domainParameterId>
                                                <locationId>Dummy</locationId>
                                                <timeSeriesType>external historical</timeSeriesType>
                                                <timeStep unit="hour"/>
                                                <relativeViewPeriod unit="hour" start="-24" end="0"/>
                                                <readWriteMode>add originals</readWriteMode>
                                                <synchLevel>1</synchLevel>
                                        </timeSeriesSet>
                                </timeSeriesSets>
                        </exportNetcdfActivity>
                        <exportNetcdfActivity>
                                <exportFile>zs0file.nc</exportFile>
                                <timeSeriesSets>
                                        <timeSeriesSet>
                                                <moduleInstanceId>Run_XBeach</moduleInstanceId>
                                                <valueType>scalar</valueType>
                                                <parameterId>H_mean</parameterId>
                                                <locationId>Dummy</locationId>
                                                <timeSeriesType>external historical</timeSeriesType>
                                                <timeStep unit="minute" multiplier="10"/>
                                                <relativeViewPeriod unit="hour" start="-24" end="0"/>
                                                <readWriteMode>add originals</readWriteMode>
                                        </timeSeriesSet>
                                </timeSeriesSets>
                        </exportNetcdfActivity>
                        <exportNetcdfRunFileActivity>
                                <description>This run file is passed as argument to XBeachPreAdapter<
/description>
                                <exportFile>run.nc</exportFile>
                                <properties>
                                        <string key="PARAM_OUTPUTFORMAT" value="netcdf"/>
                                </properties>
                        </exportNetcdfRunFileActivity>
                </exportActivities>
```

Execute activities

The next steps are the execute activities.
The first will be the pre-adapter. This program will read the run.nc input file and use the contents for instructions on which directory and files should be used to convert to the correct XBeach input format. The pre-adapter generates a log file called XBeach.log, which can be read into FEWS by coupling line patterns to FEWS log messages.
The second execute activity will be the module run. XBeach generates different log files with different meaning, in the configuration below, all line from XBerror.txt are coupled to error messages in FEWS, all lines from XBwaring.txt are coupled to info messages in FEWS and all lines in XBlog.txt are coupled to debug messages in FEWS.

execute activities

```
<executeActivities>
        <executeActivity>
                <command>
                        <className>nl.deltares.xbeach.XBeachPreAdapter</className>
                        <binDir>adapter\bin</binDir>
                </command>
                <arguments>
                        <argument>run.nc</argument>
                </arguments>
                <logFile>
                        <file>XBeach.log</file>
                        <errorLinePattern>ERROR*</errorLinePattern>
                        <warningLinePattern>WARN*</warningLinePattern>
                        <infoLinePattern>INFO*</infoLinePattern>
                        <debugLinePattern>DEBUG*</debugLinePattern>
                </logFile>
                <timeOut>99999999</timeOut>
        </executeActivity>
        <executeActivity>
                <command>
                        <executable>xbeach.exe</executable>
                </command>
                <logFile>
                        <file>XBerror.txt</file>
                        <errorLinePattern>*</errorLinePattern>
                </logFile>
                <logFile>
                        <file>XBwarning.txt</file>
                        <infoLinePattern>*</infoLinePattern>
                </logFile>
                <logFile>
                        <file>XBlog.txt</file>
                        <debugLinePattern>*</debugLinePattern>
                </logFile>
                <timeOut>99999999</timeOut>
        </executeActivity>
</executeActivities>
```

Wave spectrum conversion

"bcfile.nc" will be used to write the wave spectra into the following format:

- bcfile.txt referencing to wave spectrum files and a time of how long these should be used in each calculation step

waves.txt

```
FILELIST
3600.0      1.0 bc.time001.sp2
3600.0      1.0 bc.time002.sp2
3600.0      1.0 bc.time003.sp2
3600.0      1.0 bc.time004.sp2
3600.0      1.0 bc.time005.sp2
3600.0      1.0 bc.time006.sp2
3600.0      1.0 bc.time007.sp2
3600.0      1.0 bc.time008.sp2
3600.0      1.0 bc.time009.sp2
3600.0      1.0 bc.time010.sp2
```

- bc.time001.sp2 containing a wave spectrum
- The conversion will use either "EnDens" or "VaDens" variable from "bcfile.nc" for the values, using the unitstring specified with the variable

bc.time001.sp2

```
SWAN   1                                Swan standard spectral file, version
$   Data exported by FEWS for SWAN
$   Project:                 ;  run number:
TIME                                    time-dependent data
     1                                  time coding option
LONLAT                                  locations in spherical coordinates
     1                                  number of locations
   4.6019540   52.6194688
AFREQ                                   1/s
    25                                  number of frequencies
    0.0500
    0.0566
    0.0642
   ...
    0.7791
    0.8827
    1.0000
NDIR                                    degrees
    36                                  number of directions
  265.0000
  255.0000
  245.0000
  ...
  -65.0000
  -75.0000
  -85.0000
QUANT
     1                                  number of quantities in table
VaDens                                  id
m2/Hz/degrees                           unit
   -0.9900E+02                          exception value
20010101.000000                         date and time
FACTOR
    0.0011500214
     0     0     0     0     0     0     0     0     0     0     ...     0     0     0
     0     0     0     0     0     0     0     0     0     0     ...     0     0     0
     0     0     0     0     0     0     0     0     0     0     ...     0     0     0
     1     1     0     0     0     0     0     0     0     0     ...     0     0     1
    40    27     9     3     3     3     2     0     0     0     ...     2     9    28
   610   294    70    17    24    31    17     0     0     0     ...    67   303   626
  4916  1621   268    57   119   193   145    23     0     1     ...  1780  5295  7229
 13782  3341   415    94   287   659   803   340     0     6     ... 14433 28039 27099
 14186  2446   257    68   290   982  1874  1407     5     3     ... 52381 62127 39577
 10369  1638   231    37    90   431  1241  1598   227     0     ... 99999 80353 37089
  8225  1923   361    54    17    70   324   770   448     0     ... 77305 54223 24535
  6742  1962   381    88    21     6    30   143   211     0     ... 42814 28100 15442
  5744  2480   622   147    34     5     1     8    30     0     ... 27283 16631  9808
  4448  2618  1011   228    41     6     0     0     2     2     ... 15103  9139  6001
  2853  2150   977   217    34     5     1     0     0     0     ...  8682  5250  3507
  1818  1588   752   160    20     4     1     0     0     0     ...  4838  2895  2116
  1064   927   597   168    23     3     0     0     0     0     ...  2374  1383  1151
   517   534   423   142    23     2     0     0     0     0     ...  1182   789   561
   243   299   202    87    17     1     0     0     0     0     ...   756   329   262
   137   112    87    51    12     1     0     0     0     0     ...   243    98   120
    86    40    34    18     5     1     0     0     0     0     ...   137   117   142
    55    21     8     4     2     0     0     0     0     0     ...   137   123    87
    24    13     3     1     0     0     0     0     0     0     ...    83    74    62
     6     7     1     0     0     0     0     0     0     0     ...    58    27    16
     6     3     0     0     0     0     0     0     0     0     ...    39    20    10
```

Water level conversion

In this example "zs0file.nc" will be used to write the time dependent water levels to a file named "zs0file.txt". The first column specifies the time (meaning defined in "PARAM_TUNITS") and the second column water level. For now the adapter only supports the water level as a single boundary condition but XBeach has to possibility to also use 2 or 4 resulting in 1 or 3 extra columns.

zs0file.txt

```
0.0000000e+000 -2.2000000e-002
6.0000000e+002  2.2000000e-002
1.2000000e+003  6.4999998e-002
1.8000000e+003  1.0800000e-001
2.4000000e+003  1.5200000e-001
3.0000000e+003  1.9400001e-001
3.6000000e+003  2.3700000e-001
4.2000000e+003  2.7900001e-001
4.8000000e+003  3.1999999e-001
5.4000000e+003  3.6100000e-001
6.0000000e+003  4.0099999e-001
6.6000000e+003  4.4100001e-001
7.2000000e+003  4.7900000e-001
7.8000000e+003  5.1700002e-001
8.4000000e+003  5.5400002e-001
9.0000000e+003  5.8999997e-001
9.6000000e+003  6.2400001e-001
1.0200000e+004  6.5700001e-001
1.0800000e+004  6.9000000e-001
1.1400000e+004  7.2000003e-001
1.2000000e+004  7.5000000e-001
1.2600000e+004  7.7800000e-001
1.3200000e+004  8.0500001e-001
1.3800000e+004  8.2999998e-001
1.4400000e+004  8.5299999e-001
1.5000000e+004  8.7500000e-001
1.5600000e+004  8.9499998e-001
1.6200000e+004  9.1399997e-001
1.6800000e+004  9.3000001e-001
1.7400000e+004  9.4599998e-001
1.8000000e+004  9.5899999e-001
```

Parameter conversion

It is possible to change model parameters as defined in params.txt from FEWS. The pre-adapter will convert all run file properties starting with "PARAM_" to XBeach parameters in "params.txt". Example: params.txt.
It reads the existing "params.txt" and searches for a line starting with the specified parameter and replaces the whole line with "parameter = value" or adds a new line in the same format when the parameter was not present yet.

sample of params.txt

```
----------------------------------------------------
Grid input
nx       = 154
ny       = 70
xfile    = x.grd
yfile    = y.grd
xori     = 101627.84
yori     = 513562.63
depfile  = egmondxbeach.dep
----------------------------------------------------
Numerics input
CFL      = 0.8
eps      = 0.01
----------------------------------------------------
Time input
tstart   = 0.
tstop = 36000
----------------------------------------------------
General constants
rho      = 1025
g        = 9.81
----------------------------------------------------
Boundary condition options
zs0file  = zs0file.txt
tideloc  = 1
----------------------------------------------------
Wave calculation options
bcfile   = bcfile.txt
----------------------------------------------------
Flow calculation options
nuh      = 0.1
nuhfac   = 1.0
----------------------------------------------------
Sediment transport calculation options
facua    = 0.10
D50      = 0.0002
----------------------------------------------------
Morphological calculation options
morfac   = 10
morstart = 3600
----------------------------------------------------
Output options
outputformat = netcdf
nglobalvar = 3
tunits = seconds since 2001-01-01
```

Executing model run

The next activity will be executing the XBeach model run. This is done by running xbeach.exe in the workdir containing the model files.

module run execute activity

```
<executeActivity>
        <command>
                <executable>xbeach.exe</executable>
        </command>
        <timeOut>99999999</timeOut>
</executeActivity>
```

Import activities

The last part of the general adapter run is importing the XBeach output. xboutput.nc contains all parameter, output and grid information of the run. This can be visualized in FEWS after defining the needed parameters, location and grid. How to do this can be found here: XBeach FEWS setup

model run output import activity

```
<importActivities>
        <importPiNetcdfActivity>
                <importFile>xboutput.nc</importFile>
                <timeSeriesSets>
                        <timeSeriesSet>
                                <moduleInstanceId>Run_XBeach</moduleInstanceId>
                                <valueType>grid</valueType>
                                <parameterId>H_max</parameterId>
                                <locationId>Dummy</locationId>
                                <timeSeriesType>external historical</timeSeriesType>
                                <timeStep unit="hour"/>
                                <readWriteMode>add originals</readWriteMode>
                        </timeSeriesSet>
                </timeSeriesSets>
        </importPiNetcdfActivity>
</importActivities>
```