Scenario and Script Database - SSD

SSD folder structure

The scenario/script database for the Water Coach is a directory containing an arbitrary number of scenarios/scripts. A scenario consists of a FEWS database (the so-called local data store) that contains measurement data and model data, and one or more script directories. Each one of these script directories uses the same scenario (i.e. the local data store). A script is the story line and consists of all events that happen during the game. Each subdirectory of the scenario directory that contains a file named `script_config.xml` is considered a script and will automatically appear in the list of scripts that can be selected by the user of the Water Coach. The following files/directories must be part of a scenario /script directory:

- A scenario directory must contain a directory named `localDataStore` which holds the pre-loaded FEWS database. When a scenario/script is started in the Water Coach, this directory is copied to the FEWS region directory.
- A scenario directory must contain one or more script directories.
- A script directory must contain a script configuration file named `script_config.xml`.
- A script directory may contain an arbitrary number of files that are associated with events specified in the script configuration.

The Water Coach adds a log-directory with log-files to the script directory that was chosen by the user. A log-file is created for each scenario/script that is played and is added (with a time stamp) to the appropriate directory in the scenario/script database.
On the right is an example of a database containing two scenarios. One scenario contains two scripts, the other scenario contains one script.

```
ScenarioScriptDatabase
|-- storm surge March 10-13
|   |-- localDataStore
|   |   `-- local.fdb
|   |-- novice
|   |   |-- script_config.xml
|   |   |-- exercise1_EN.txt
|   |   |-- exercise1_NL.txt
|   |   |-- start_session_EN.txt
|   |   |-- start_session_NL.txt
|   |   |-- telephone_call_EN.txt
|   |   |-- telephone_call_NL.txt
|   |   |-- weather_map_March12_00hr.pdf
|   |   |-- weather_map_March12_12hr.pdf
|   |   `-- weather_map_March13_00hr.pdf
|   |-- advanced
|   |   |-- script_config.xml
|   |   |-- telephone_call.txt
|   |   `-- weather_map.pdf
`-- another storm surge
    |-- localDataStore
    |   `-- local.fdb
    `-- script1
        |-- script_config.xml
        |-- file1.pdf
        `-- file2.pdf
```

SSD folder structure for different (older) versions of a Delft-FEWS application (aka WaterCoach meta-application)

If different (older) WaterCoach scenarios are associated with different (older) configuration and/or modules, the corresponding Config folder and/or Modules folder can be placed in the WaterCoach scenario folders as well.

The user experience is the same as before, but under the hood more things happen. After selecting a scenario and script, Delft-FEWS will restart as always, but will now not only copy the LDS, but also unzip and replace the Config (and Modules) folder in the $REGION_HOME$

To let WaterCoach Meta-application work correctly, please follow these rules:

- use only the names Config.zip, and Modules.zip,
- if any scenario contains a Config.zip, all scenarios need to contain a Config.zip,
- if any scenario contains a Modules.zip, all scenarios need to contain a Modules.zip,
- WaterCoachDisplay.xml: set copyLocalDataStore=true for all Config.zip versions
- when the global.properties are not the same for all, place them in the RootConfigFiles folder, and use the same name for all global properties files (including your stand alone application which is started by the user).

SSD including different (older) versions of the Config and Modules folder

```
ScenarioScriptDatabase
|-- scenario1
|   |-- localDataStore
|   |   `-- local.fdb
|   |-- script1
|   |   `-- script_config.xml
|   |-- Config.zip
|   |-- Modules.zip
`-- scenario2
    |-- localDataStore
    |   `-- local.fdb
    |-- script1
    |   `-- script_config.xml
    |-- Config.zip
    `-- Modules.zip
```

SSD folder structure for multiple Delft-FEWS applications

It is also possible to group several scenarios/scripts (that belong to different Delft-FEWS configurations for example) by using an extra directory level.

In the application configuration, a reference to the appropriate scenario/script database can be set, for example `ScenarioScriptDatabase/River`.

Minimize size of LDS through changes to Delft-FEWS config

To minimize the size of the localDataStore, one could separate the import of maps and timeseries into separate modules. This allows for a lower frequency for the import of maps. However, when this is applied, an additional change should be made to the Delft-FEWS configuration (see also application configuration). Also make sure to incorporate the changes to the moduleInstanceId's for the separate import of timeseries and maps in the Delft-FEWS configuration of the filters, spatial display, etc.

Remove information concerning the WaterCoach session from the SSD

When a LDS (localDataStore) scenario is made, the LDS should not contain any information concerning a WaterCoach session. Informatio n on a selected scenario, script and time is stored in the LDS whenever we start a WaterCoach session. Sometimes users make a LDS for a scenario when they are in a Watercoach session. It is advised to always use a LDS outside of a WaterCoach session as FEWS is storing the session information in a LDS table. It is therfore advised to remove information on an uncompleted WaterCoach session from the LDS before it is copied to the scenario folder. This can be done with the Explorer **F12 menu   P database > delete WaterCoach session info**. This should be done when the LDS is opened in a Delft-FEWS Stand Alone. Follow the procedure:

1. Copy the Watercoach Scenario LDS to a Stand Alone
2. Start the Stand Alone
3. Press F12 and select the new option to delete the WaterCoach Info from the database table
4. Close the Stand Alone
5. Copy the LDS to the WaterCoach folder

SSD for multiple Delft-FEWS applications

```
ScenarioScriptDatabase
|-- Sea
|   |-- scenario1
|   |   |-- localDataStore
|   |   |   `-- local.fdb
|   |   |-- script1
|   |   |   `-- script_config.xml
|   |   `-- script2
|   |       `-- script_config.xml
|   `-- scenario2
|       |-- localDataStore
|       |   `-- local.fdb
|       `-- script1
|           `-- script_config.xml
`-- River
    |-- scenario1
    |   |-- localDataStore
    |   |   `-- local.fdb
    |   `-- script1
    |       `-- script_config.xml
    `-- scenario2
        |-- localDataStore
        |   `-- local.fdb
        |-- script1
        |   `-- script_config.xml
        `-- script2
            `-- script_config.xml
```