

# THE MIKE CUSTOMISED PLATFORM

## Job Manager

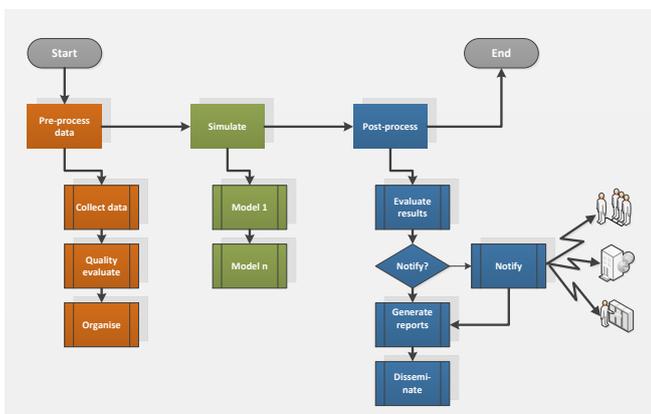
### BATCH PROCESSING

Many Water Information Management-related systems require possibilities for flexible and extensible batch processing. These notably include operational systems which (on a scheduled basis) calculate prognostics variables. The latter are related to flood forecasting, optimal gate structure operations and so on.

Batch processing is often very computation intensive. It therefore requires powerful computers with multiple CPU cores and large internal memory. Batch processing usually includes execution of multiple tasks like:

- Pre-processing tasks for data preparation
- Running simulation software
- Condensing simulation output in reports
- Sending gate operation instructions
- Notifying operators and end-users about possible hazards

For example, it is not rare that real-time optimisation batch processes execute hundreds of simulations to reach an optimal solution. This calls for the batch processing to be moved from the user's workstations to dedicated simulation servers.



Sample job workflow

Different solutions require different batch processing – data is typically collected from several unique source systems. Data validations and preparation rules vary, as do the models and modelling software. Post-processing and notification rules

### SUMMARY

#### CLIENT

The Job Manager provides flexible and extensible batch processing to the MIKE CUSTOMISED Platform

#### CHALLENGE

Water information management systems require:

- Batch execution facilities for running recurrent forecast models
- Abilities for 'drag-and-drop'-based configuration of jobs
- Possibilities for distributing batch processing across the network
- Ways of extending the tasks that can be included within a job

#### KEY FEATURES

- Local and remote execution of jobs
- Large number of standard tasks covering the vast MIKE CUSTOMISED functionality
- Scheduling jobs for recurrent execution
- Uses MIKE CUSTOMISED scripts as job tasks
- Leveraging a standard workflow engine
- Extensive documentation for both building jobs and programming solution specific tasks

also differ to a large extent. It is therefore necessary that a batch system is flexible in setting-up and does not require complex programming. In order to be cost efficient it will be possible to 'point-and-click' define the content of the batch processes.

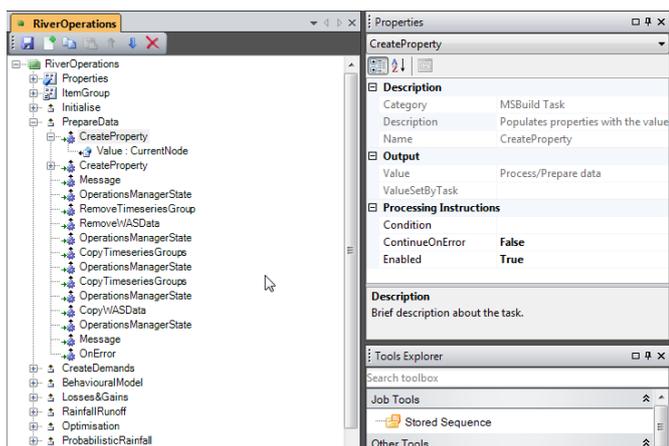
## THE JOB MANAGER

With the MIKE CUSTOMISED Platform, batch processing is managed through the Job Manager component. This component provides a rich user interface for defining and setting-up 'jobs' - the Job Manager jargon for batch processing. A job is an XML structure that contains a list of tasks that are executed in a pre-defined order. The tasks typically execute well-defined actions such as:

- Importing a time series to the Platform database
- Running a simulation scenario
- Performing a GIS zonal statistics analysis

The user defines the execution order of the tasks via 'targets', where a target simply is a list of tasks. Execution-wise, targets can depend on each other and thus form a semi-hierarchical structure.

Users define their jobs by dragging tasks from a task palette onto a job view in a workflow-oriented manner. Task outputs can be used as input for tasks executed later. The individual tasks are configured in a task property control editor. Solution implementing projects can augment the standard list of task in several ways. It can be done by leveraging the MIKE CUSTOMISED built-in scripting capabilities. Alternately, you can also programme a custom .NET component and have that appear as a task complementing the standard tasks.

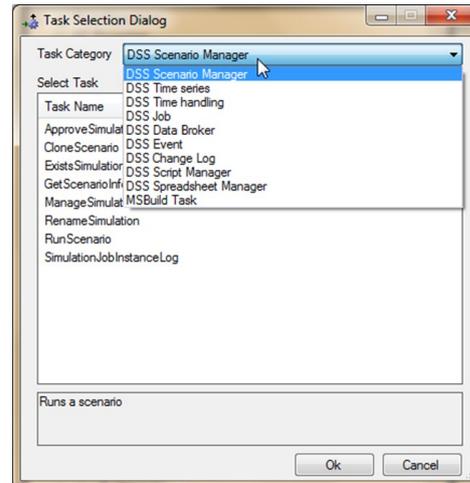


### Defining jobs

Programming job tasks are extremely simple and typically do not require more than 30 to 40 lines of code.

Contact: MIKE CUSTOMISED - [mikecustomised@dhigroup.com](mailto:mikecustomised@dhigroup.com)  
For more information visit: [www.dhigroup.com](http://www.dhigroup.com)

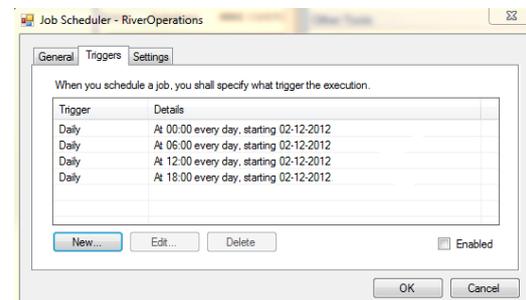
MIKE CUSTOMISED comprises extensive documentation on how to work with jobs and how to program tasks.



Selection of job tasks

## FEATURES

- Declarative definition of jobs through dragging tasks from a palette of standard tasks. Solution implementing projects can add their own tasks to the palette
- Configuration of tasks through input and output properties
- MIKE CUSTOMISED scripts can be executed as tasks, providing an easy and cost efficient way to embed solution-specific functionality in the job execution



Scheduling of jobs

- Jobs can be executed directly or through a scheduling mechanism. This can be done locally or remotely on any computer on the network configured for job execution
- Scheduled execution includes both one-time and recurrent execution
- Job execution (directly or scheduled), generates a log with extensive information on the job processing