FP7 Grant Agreement N° 312450

CIPRNet

Critical Infrastructure Preparedness and Resilience Research Network

Project type: Network of Excellence (NoE)

Thematic Priority: FP7 Cooperation, Theme 10: Security

Start date of project: March 1, 2013 Duration: 48 months

D9.2 Draft training material OpenMI

Due date of deliverable: 31/12/2013
Actual submission date: 5/03/2014

Revision: Version 1.1

Stichting Deltares, The Netherlands

Project co-funded by the European Commission within the Seventh Framework Programme (2007–2013)		
Dissemination Level		
PU	Public	x
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Author(s)	Annette Zijderveld (Deltares) Andreas Burzel (Deltares)
Contributor(s)	Bernhard Becker (Deltares)

Security Assessment	**Erich Rome**
Approval Date	25.2.2014
Remarks	No issues.

The project CIPRNet has received funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no 312450.

The contents of this publication do not reflect the official opinion of the European Union. Responsibility for the information and views expressed herein lies entirely with the authors.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

Acronym	Explanation
CI	Critical Infrastructure
CIP	Critical Infrastructure Protection
CIPRNet	Critical Infrastructure Preparedness and Resilience Research Network
OpenMI	Open Modelling Interface
RTC Tools	Real Time Control Tools
UIC	International Union of Railways

1 Introduction – Rationale of this document

CIPRNet training activities include four major training events. These consist of one internal training event and three Master Classes for external participants. The internal training event, dubbed "Edition 0", has been held in Delft on February 3 and 4, 2014, on the premises of Deltares.

The first training event of CIPRNet, Edition 0, focused on basic questions of simulating complex systems, interdependencies of critical infrastructures and different methods of federated simulations. OpenMI (Open Modelling Interface) is one example of a mature technology to combine different simulation models for complex systems. A major part of Edition 0 consisted of an introduction to OpenMI combined with "hands on" exercises for familiarising with the OpenMI approach and with some of the models and simulations using OpenMI.

This document contains the draft training material on OpenMI, as it was used for Edition 0. More training activities (lectures, courses, Master Class) on OpenMI are planned, so this material will be reused and developed further. The first Master Class is planned for April 24–25 in Paris, France. In order to better understand the context of the training material, we include the programme of edition 0 and a brief characterisation of the target audiences of the training events. The general training plan is described in D9.1 Training Plan [D9.1] of CIPRNet.

For a report on the entire "Edition 0" training event, see the forthcoming deliverable D9.51. For more information on OpenMI, please check the OpenMI website (openmi.org)

1.1 Target audience

The target audience of the Edition 0 training event were CIPRNet staff members and interested colleagues, so mostly researchers, academic staff, and a few end user representatives (from partner UIC).

The target audience of the forthcoming Master Class are:

- Technicians /Researchers on CIP,
- CI operators and Public Authorities,
- Crisis management team members,

who

- are working in the field of disaster management and critical infrastructure,
- want to enhance their knowledge on how behaviour of complex systems can be simulated,
- are interested on the design and the functionalities of a federated simulator of CI, and
- have to deal with cascading effects of failure of critical infrastructures in a crisis situation/ decision making process.

The first Master Class of CIPRNet is set up to concentrate on simulation concept and coupling methods, such as OpenMI.

1.2 Topics of the CIPRNet training event on Federated modelling & simulation with focus on OpenMI

The complete program is described in the CIPRNet Training Plan [D9.1]. To understand the total scope of the Training Edition "0", the main topics are listed here:

Programme of day 1:

Topic	lecturer	affiliation
Introduction to Modelling and Simulation of systems	Mohamed Eid	**CEA**
Dependencies and interdependencies	Roberto Setola	**UCBM**
Holistic modelling	Roberto Setola	**UCBM**
Topological properties of complex networks and their relevance in functional and vulnerability assessments of Critical Infrastructures	Vittorio Rosato	**ENEA**
Geomatics	Maurizio Pollino	**ENEA**
Introduction to Federated Simulation	Wim Huiskamp	**TNO**
Modelling and Simulation Techniques for Critical Infrastructure Protection	Andrij Usov	**Fraunhofer**
Internal review and evaluation		

Programme of day 2:

Topic	lecturer	institute
Verification and Validation methods	Jeroen Voogd	**TNO**
Simulations of CI - relevant examples	Eric Luiijf	**TNO**
Introduction to OpenMI	Andreas Burzel	**Deltares**
OpenMI behind the scenes: how to migrate my own code to OpenMI compliance	Bernhard Becker	**Deltares**
Hands-on training: my first OpenMI composition	Bernhard Becker	**Deltares**
Internal review and evaluation		

2 Training material on OpenMI

2.1 Lecture material

The introduction to OpenMI is given as a plenary lecture. The additional teaching material (tutorial) as provided to the audience is divided in the following chapters:

1) Introduction
2) Model coupling and conjunctive modelling
3) The OpenMI standard
4) Example: coupled flow simulation and control

For the lecture two presentations have been used.

2.2 Exercises

In the exercise, the audience is experiencing the example of combining a simple hydrological model (river branch) with a rule-based model for steering a wire: RTC tools (Figure 1). This example shows in well-defined steps how to connect these two models based on the OpenMI concept.

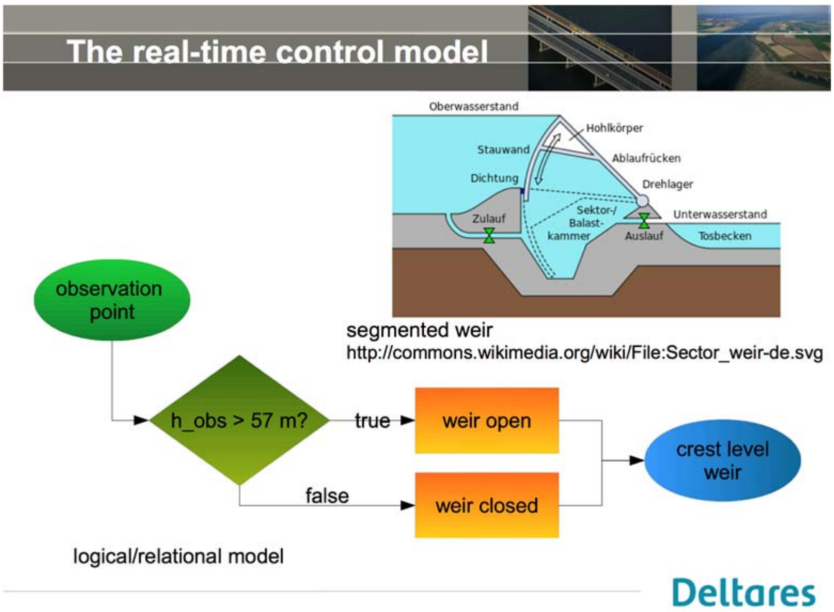


Figure 1: Real time control model schematisation (source: OpenMI Training material for CIPRNet)

3 Learning goal

This tutorial gives an introduction on conjunctive modelling with OpenMI. The functional principle of OpenMI is explained with the help of two water-related models. The first objective of this course is to provide a general introduction into model coupling.

Learning goal is to know basics of different coupling methods and different modes of process interaction modelling. The second section provides technical explanation of the OpenMI standard. Students learn what an OpenMI compliant component is and learn how the data exchange works. In the third section students build their own OpenMI composition. Students learn how to load models and how to configure the connection between models.

4 Presentation of the material

The tutorial of the OpenMI training course is included in Appendix A. The Powerpoint presentations are included in Appendix B. All other material, such as the software modules for the training case, are available on the CIPRNet server BSCW.

5 References

[D9.1] UCBM: D9.1 Training Plan, CIPRNet deliverable D9.1, Rome, February 2014

[D9.51] UCBM: Periodic training event internal, CIPRNet deliverable 9.51, Rome, to appear, April 2014

[OpenMI] OpenMI organisation: http://www.openmi.org

6 Further Reading

OpenMI Association, see www.openmi.org:

- information about OpenMI and the OpenMI Association
- download and documentation of OpenMI 1.4 and 2.0

Deltares OpenMI public wiki, see publicwiki.deltares.nl:

- OpenMI documentation
- how-to tutorials and manuals

Becker, B., Talsma, J.: On the external and iterative coupling of multiple open channel flow models with OpenMI. Revista de Ingenier´ıa Innova, Vol. accepted for publication (2014).

Becker, B. P., Schüttrumpf, H.: An OpenMI module for the groundwater flow simulation programme Feflow. Journal of Hydroinformatics, Vol. 13 (1), pp. 1–13 (2010), ISSN 1464-7141, doi:10.2166/hydro.2010.039. URL http://www.iwaponline.com/jh/013/jh0130001.htm.

Becker, B. P. J.: Zur gekoppelten numerischen Modellierung von unterirdischem Hochwasser. Dissertation, RWTH Aachen, Fakultät für Bauingenieurwesen, Aachen (2010). In German, URL http://d-nb.info/100850209X/34.

Becker, B. P. J., Forberig, S., Flögel, R., Schüttrumpf, H., Köngeter, J.: On the Determination of Groundwater Levels for Hazard Maps of Groundwater Head Rise Induced by High Water. Wasserwirtschaft, (12), pp. 10–16 (2011), ISSN 0043-0978.

Becker, B. P. J., Talsma, J., Gao, Q., Ruijgh, E.: Coupling of multiple channel flow models with OpenMI. Proceedings of 10th International Conference on Hydroinformatics, Hamburg, Germany (2012c), ISBN 978-3-941492-45-5.

Deltares 2013: RTC-Tools a toolbox for real-time control of hydraulic structures (2013). Published: Deltares, URL http://oss.deltares.nl/web/rtc-tools.

Gijsbers, P., Gregersen, J., Westen, S., Dirksen, F., Gavardinas, C., Blind, M.: OpenMI Document Series: Part B – Guidlines for the OpenMI. IT Frameworks (HarmonIT), version 1.0 edn. (2005). URL http://www.openmi.org/openminew/documents/B_Guidelines.pdf.

Moore, R., Gijsbers, P., Fortune, D., Gregersen, J., Blind, M.: OpenMI Document Series: Part A – Scope for the OpenMI. Version 1.0 edn. (2005).
URL http://www.openmi.org/openminew/documents/A_OpenMI_Scope.pdf.

Werner, M., Schellekens, J., Gijsbers, P., van Dijk, M., van den Akker, O., Heynert, K.: The Delft-FEWS flow forecasting system. Environmental Modelling & Software, Vol. 40, pp. 65–77 (2013), ISSN 13648152, doi: 10.1016/j.envsoft.2012.07.010.
URL http://linkinghub.elsevier.com/retrieve/pii/S1364815212002083.

Appendix A: OpenMI Tutorial

Conjunctive modelling with OpenMI

Critical Infrastructure Preparedness and Resilience Research Network (CIPRNet)
Work Package 9 training material

Bernhard Becker* Andreas Burzel†

February 3, 2014

1 Introduction

This tutorial gives an introduction on conjunctive modelling with OpenMI. The functional principle of OpenMI is explaned with the help of two water-related models. The first objective of this course is to provide a general introduction into model coupling. Learning goal is to know basics of different coupling methods and different modes of process interaction modelling. The second section provides technical explanation of the OpenMI standard. Students learn what an OpenMI compliant component is and learn how the data exchange works. In the third section studens build their own OpenMI composition. Students learn how to load models and how to configure the connections between models.

2 Model coupling and conjunctive modelling

2.1 What is a model?

> "A model should be made as simple as possible, but not simpler."
> (after Albert Einstein, 1879 - 1955)

Following Konikow & Bredehoeft (1992) we use the following definitions:

A *model* is a representation of a real system or process. A *conceptual model* is a hypothesis for how a system or process operates. *Mathematical models* are abstractions that replace objects, forces, and events by expressions that contain mathematical variables, parameters and constants. *Deterministic models*, also called physics-based models, are based on the conservation of mass, momentum and energy. Deterministic models often require the solution of differential equations for certain boundary and initial conditions. A mathematical model, or, more in particular, a numerical algorithm to solve differential equations, implemented into computer

*Deltares, P.O. Box 177, 2600 MH Delft, the Netherlands, bernhard.becker@deltares.nl, +31 6 5241 6736
†Deltares, P.O. Box 177, 2600 MH Delft, the Netherlands, andreas.burzel@deltares.nl

code is called a *computer model*. This computer model can also be considered as a *generic model*. When model parameters, boundary conditions and grid definitions for a generic model are specified to represent a particular geographic area, we obtain a *site-specific model*, including model data and software. A *synthetic model* represents a fictitious site, often used to illustrate or analyse a certain process.

A computer model usually consists of a graphical user interface part and a computational core that solves the partial differential equation system.

Flow processes are often described mathematically by partial differential equations. These equations cannot be solved analytically. The numerical solution requires a grid (mesh) that represents the modelling area. The solution of differential flow equations requires a full definition of the boundary of the modeling area, the so-called boundary conditions. In addition, internal boundary conditions like sources and sinks can be defined. Transient flow problems require initial conditions for the whole modeling area grid. A set of boundary conditions and initial conditions.

2.2 What is conjunctive modelling?

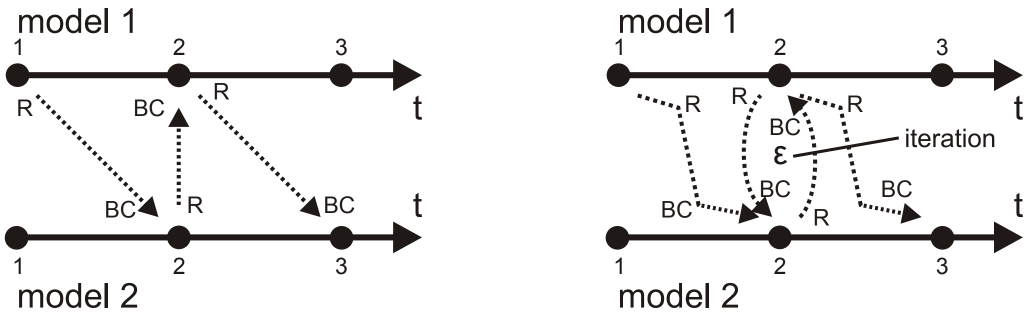
Conjunctive modelling means to link site-specific models in such a way that the interaction processes between the models are modelled on a time-step basis, the processes are coupled. Models can be linked by model coupling or uncoupled. Coupling means data transfer in two directions. Simulation results of the first model have an impact on the second model and vice versa. Uncoupled conjunctive modelling means data transfer in one direction only: simulation results have an impact on the second one, but the simulation of the second model has no feedback impact on the first one. Coupled models must exchange data during runtime on a time step basis, while uncoupled model linking can also be done manually.

Three levels of model coupling according to Morita & Yen (2002) and Morita & Yen (2000):

- simultaneous total coupling
- alternating iterative coupling
- externally coupled

External coupling means data exchange once per time step. Results from one model are used as boundary conditions in the other one and vice versa (see Fig. 1a). This is the lowest level of model coupling. Also called time-lagged approach (Fairbanks et al., 2001; Huang & Yeh, 2006), this approach is the least accurate one, because it contains inherent mass balance and momentum balance errors. But this approach is the certainly most applied one, because it is easier to implement than the other two, and often sufficient.

Iterative coupling means to exchange data between models not only once per time step, but to iterate the exchange of data until a certain convergence criterion is achieved (see Fig. 1b). Consequently, mass balance errors and momentum errors are basically smaller than for external coupling. But this method is more difficult to implement and more computational expensive.



(a) external coupling (b) iterative coupling

Figure 1: Functional principle of external coupling and iterative coupling of two models (after Becker, 2010). Abbreviations: R: result, BC: boundary condition, t: time, ε: convergence criterion.

Simultaneous coupling is the highest level of model coupling. It means to represent different processes, including the interactions, in one equation system. However, the simultaneous solution requires equal time stepping for all coupled processes, and the equations should be of the same type to make it efficient.

OpenMI supports iterative coupling and external coupling. Morita & Yen (2000) and Becker & Talsma (2014) discuss numerical aspects of these model coupling approaches.

As uncoupled approach we consider the successive execution of two model simulations where the first model produces boundary conditions for the second one. A feedback from the second one to the first is not incorporated. An uncoupled conjunctive modelling can be realized by simple exchange of input and output files between models. The easiest way to implement such an uncoupled conjunctive modelling is to implement simulation results from one model as boundary conditions for the second model manually or script-based. OpenMI can help to improve efficiency for uncoupled conjunctive modelling as shown by Becker & Schüttrumpf (2010). More advanced approaches incorporate a data integration platform like Delft-FEWS (Werner et al., 2013).

2.3 Task

Your task is the development of simple process chains with practical relevance for homeland security.

1. Compose a scenario including different possibilities of critical infrastructure failure.
2. Identify models that can represent the relevant processes
3. Draw a sketch of models and their interactions.
4. Discuss benefits of model coupling for your setup against uncoupled modeling.
5. Discuss alternative setups.

Table 1: Relevant processes and corresponding models

no.	process	model
1	rainfall-runoff	hydrological model
2	river flow	1D open channel flow model
3	dike breach	dike breach model
4	hinterland flooding	two-dimensional flood model
5	groundwater head rise (subsurface flood)	groundwater model

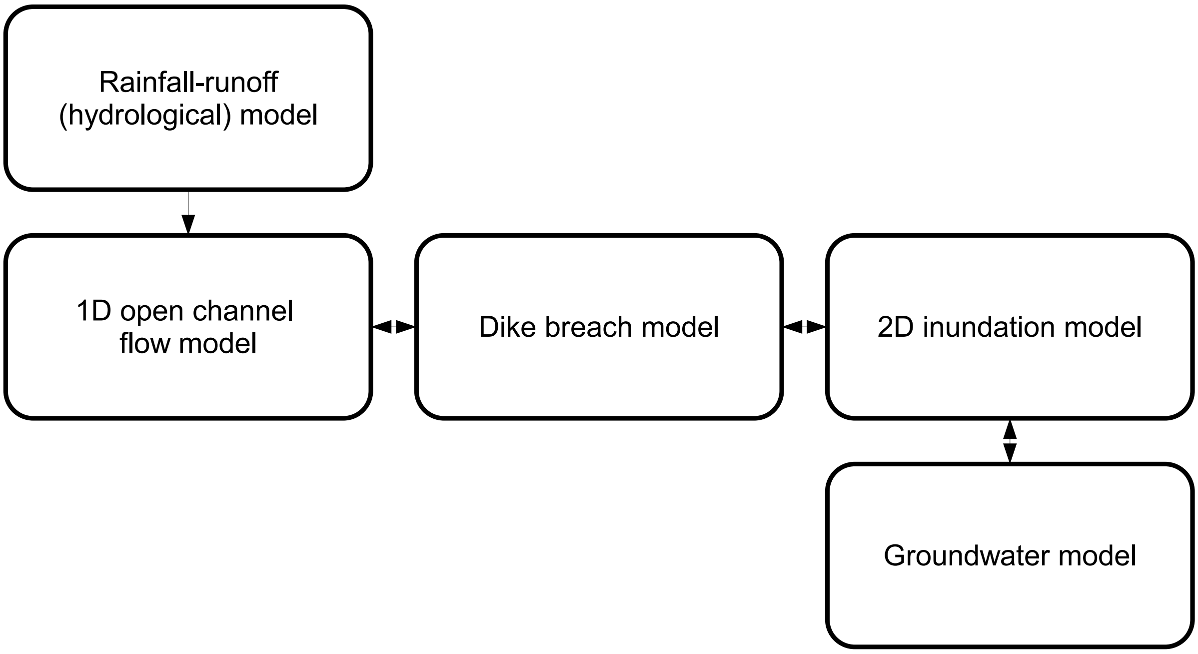


Figure 2: Model coupling for process interaction modelling. Arrows indicate data exchange between models to represent process interaction.

An example solution for task 1 is given with the following list:

– Heavy rainfall causes high water in a river.

– High water in a river causes dike breach due to overtopping.

– The dike breach causes inundations of the hinterland.

– From the inundated areas water infiltrates into the subsurface and causes groundwater head rise.

– Rising groundwater levels create uplift forces on a road tunnel and flows cellars with information technology installation.

The corresponding models (task 2) are given in Tab. 1. The model interaction is sketched in Fig. 2.

The benefit of model coupling (task 4) is

– Feeding the rainfall-runoff model with a rainfall scenario produces results without manual data transfer between the models.

– River flow, dike breach and inundation are processes that interact with each other. Uncoupled modelling would violate the mass balance of water.

– The infiltration of water from inundated areas into groundwater is an interaction process which cannot be modelled uncoupled.

– The model chain provides information that can be used to identify endangered critical infrastructure.

Alternative setups 5:

– A connection between the river model and the groundwater model adds bank interaction to the system model.

– Interactions between river model, dike breach model and two-dimensional flow model could be made uni-directional to trade-off accuracy against performance.

– A geotechnical model for failure mechanisms due to uplift forces can be added to the modelling chain.

3 The OpenMI standard

3.1 Introduction

The OpenMI standard defines an interface that allows time dependent models to exchange data at runtime (Moore et al., 2005). Model components that comply with the OpenMI standard can, without any programming, be coupled to OpenMI modelling systems (Gregersen et al., 2007; Moore & Tindall, 2005). The OpenMI environment provides tools that facilitate the migration of legacy code. This grants a high acceptance of coupled models by users, because they can use their already existing models in coupled simulations.

Beside the standard interface specification, the OpenMI-association also provides the OpenMI environment. This is a software that assists in the implementation of the OpenMI standard. It contains compiled .NET assemblies and the source code of all packages and their documentation (Moore et al., 2005).

An OpenMI system is a software system where different OpenMI compliant components are connected to a coupled modelling system. The OpenMI data exchange is based on a pull-driven request-reply mechanism. One component, for example a site-specific model, requests data needed for the own computation from another component. Components can be connected in different manners:

– unidirectional connection

– bidirectional connection

– iterated connection

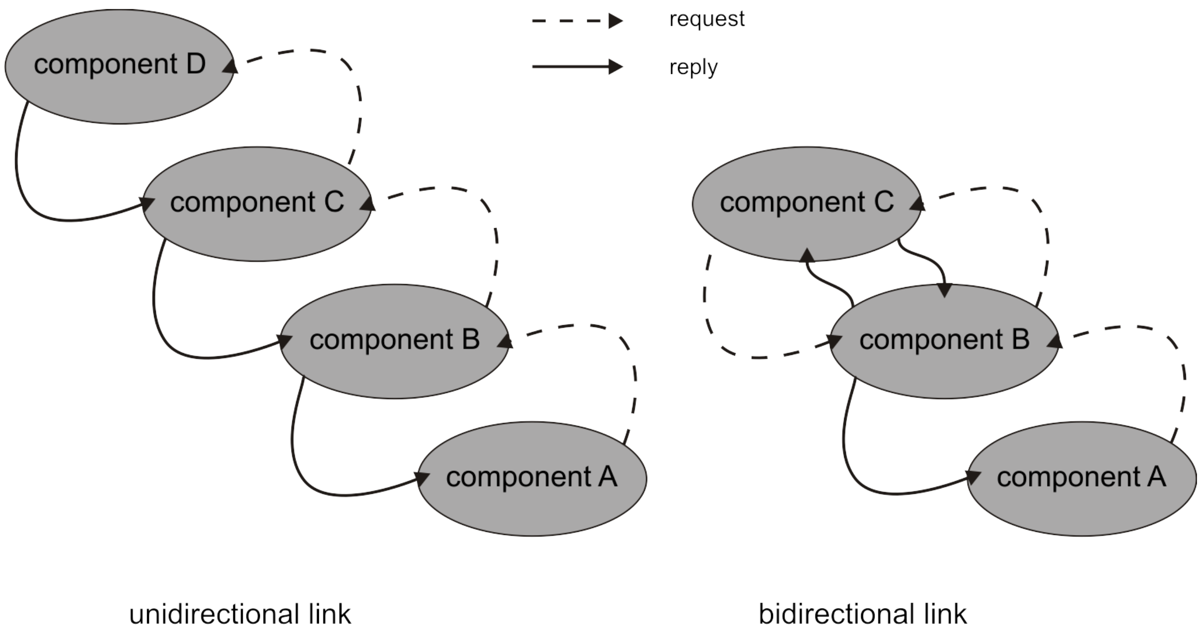


Figure 3: Different connection layouts with the request-reply mechanism (after Gregersen et al., 2007)

According to section 2.2, unidirectional connection supports the uncoupled approach, the bidirectional connection is for external coupling and the iterated connection helps to realize an iterated coupling with OpenMI. The simultaneous solution cannot be achieved with OpenMI.

In Figure 3, different layouts of pull-driven request-reply connections are shown. For the unidirectional chain, component A requests B for data. In order to response, it needs data from another component itself and requests C for data, which again requests data from component D. D is at the end of the chain and performs its computation first and then answers C. C is now able to compute and answers the request of component B afterwards. B now calculates with the data from C and is able to response on the request of A. For the bidirectional connection example, component A requests data from B. B needs data from component C. To fulfil this request, C needs data from B. Because B waits for data from C itself, it gives a guess to C. C computes with this guess and can now response to B. B is now able to compute and to reply to the request of A.

Both examples show, that one component must initialize the computation with a request to define which component shall compute first. That is why each OpenMI system contains an element which triggers the simulation. For the bidirectional connection, simulation results may differ depending on which component computes first and gives a guess. Gregersen et al. (2007) call this coupling semi-explicit, because the results of one component are based on a guess, but the results of the other component are based on a calculation. The iterative connection is an advanced bidirectional connection. In the example of 3 (right side), components B and C would adjust their reply values iteratively as long as an accuracy criterion is fulfilled. A connection between model components consists of links. A link is defined between an output exchange item and an input exchange item of two different model components, respectively. An exchange

Table 2: omi-file for a Sobek model

```
<?xml version="1.0"?>
<LinkableComponent xmlns="http://www.openmi.org"
  Type="DeltaShell.OpenMIWrapper.DeltaShellOpenMILinkableComponent"
    Assembly=".\bin\DeltaShell.OpenMIWrapper.dll">
  <Arguments>
    <Argument Key="DsProjFilePath" ReadOnly="true"
      Value=".\SobekRiverFlowModel\Magdeburg3.dsproj" />
    <Argument Key="ModelName" ReadOnly="true" Value="integrated model" />
    <Argument Key="ResultingDsProjFilePath" ReadOnly="true"
      Value=".\SobekRiverFlowModel\MagdeburgRTC.dsproj" />
    <Argument Key="SplitSpecificElementSets" ReadOnly="true"
      Value="CalcPoints;Laterals;Structures;Measurements" />
  </Arguments>
</LinkableComponent>
```

item defines a simulation time related quantity and its unit for an ordered set of elements, e. g. a single node number, a node coordinate, or lines, polygons or polyhedrons. Input exchange items usually form boundary conditions in an OpenMI compliant model component, while output exchange items are mostly simulation results.

3.2 OpenMI composition components

The omi-file contains information about one single OpenMI compliant component:

- Where is the DLL with the computational core and OpenMI-Interface?
- Where is the working directory with input files?
- Anything else like command line arguments or specific settings?

The omi-files are structured in xml. The omi-file must be created by the modeller. An example of an omi-file is given in Table 2.

3.3 Connections

The opr-file defines how OpenMI components are connected within an OpenMI composition and contains runtime information:

- Which components are part of the composition (reference to omi-files and trigger component)?
- Which connections are defined between components?
- Details of the connections (what and where)?
- Simulation period.

The opr-file is created by the OpenMI configuration editor, but can be modified by the modeller. Like the omi-file, the opr-file is structured in xml. Table 3 gives an example for an opr-file.

Data exchange is defined with the help of so-called OpenMI exchange items. An exchange item consists of

- an element set, for example one or multiple node numbers, and
- quantity and unit, for example water level in metres.

3.4 Example cases of conjunctive modelling with OpenMI

Example cases of conjunctive modelling with OpenMI under contribution of the authors of this document are given in the following list:

- Generation of boundary conditions for a transient dam seepage scenario (Becker & Schüttrumpf, 2010).
- Modelling of surface-subsurface interactions, i. e. bank storage and vertical infiltration from a flooded area) (Becker et al., 2011).
- Coupling of an open channel flow model with a pump model to design a large pump station (Becker et al., 2012a).
- Coupling of models of the same type to bridge administrative boundaries (Becker et al., 2012c; Becker & Gao, 2012).
- Integration of different hydrological processes (Schellekens et al., 2012).
- Real-time control of hydraulic structures in open channel flow models to model the human interactions in a water system Becker et al. (2012b); Becker (2013).

4 Example: coupled flow simulation and control

4.1 Study area and modelling objective

The study area is a part of the Elbe river at Magdeburg (Germany). An overview is given in Figure 4. The modelling objective is to manage the river in such a way that the water levels remain below the flood warning level. Beside the city of Magdeburg, critical infrastructure might be affected in case of flooding.

- the main station
- two railway junctions.

Table 3: opr-file for an OpenMI composition with a Sobek model and an RTC-Tools model

```xml
<guiComposition version="1.0">
  <models>
    <model omi="d:\OpenMIcourse\OpenMICoursePackageElbe\RtcTools.omi"
      rect_x="195" rect_y="113" rect_width="100" rect_height="51" />
    <model omi="d:\OpenMIcourse\OpenMICoursePackageElbe\ElbeSobek.omi"
      rect_x="52" rect_y="112" rect_width="100" rect_height="51" />
    <model omi="Oatc.OpenMI.Gui.Trigger"
      rect_x="196" rect_y="30" rect_width="100" rect_height="51" />
  </models>
  <links>
    <uilink
      model_providing="integrated model"
      model_accepting="RtcTools_ModelId">
      <link id="2"
        source_elementset="ObservationPoint1"
        source_quantity="Water level (op)"
        target_elementset="Water level (op)@ObservationPoint1"
        target_quantity="Water level (op)" />
    </uilink>
    <uilink
      model_providing="RtcTools_ModelId"
      model_accepting="integrated model">
      <link id="4"
        source_elementset="Crest level (s)@Weir1"
        source_quantity="Crest level (s)"
        target_elementset="Weir1"
        target_quantity="Crest level (s)" />
    </uilink>
    <uilink
      model_providing="integrated model"
      model_accepting="Oatc.OpenMI.Gui.Trigger">
      <link id="6"
        source_elementset="Node001"
        source_quantity="water_level"
        target_elementset="TriggerElementID"
        target_quantity="TriggerQuantityID" />
    </uilink>
  </links>
  <runproperties
    listenedeventtypes="11111111111"
    triggerinvoke="01/26/2000 00:00:00"
    runinsamethread="0" showeventsinlistbox="1"
    logfilename="CompositionRun.log" />
  <mainForm width="360" height="277" />
  <sdk>
    <smartbuffer maxnumberoftimes="0" />
  </sdk>
</guiComposition>
```

Figure 4: Study area (taken from `www.maps.google.com`)

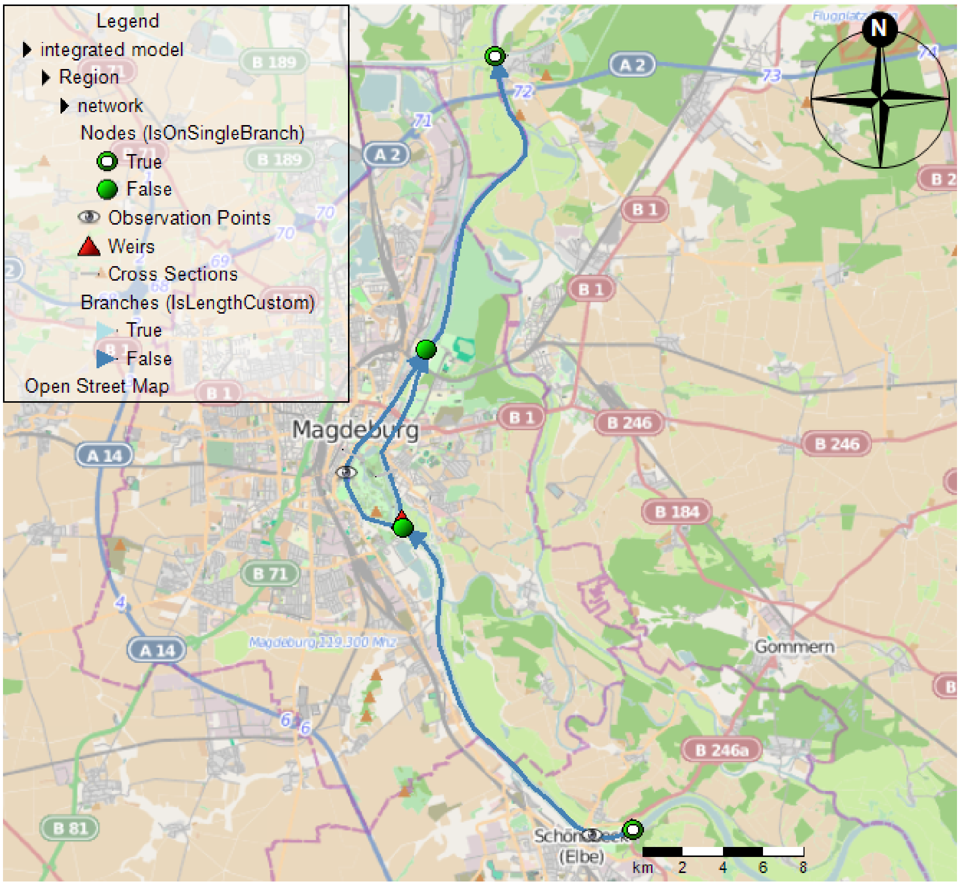


Figure 5: Sobek open channel flow model network. Water flows from south to north.

4.2 Approach

The relevant processes are

- open channel flow in the section of the river Elbe and
- human operations in the river system.

We use two models to represent these processes:

- a Sobek open channel flow model for the Elbe river and
- a real-time control model RTC-Tools to represent the human operations in the water system.

4.3 The Sobek open channel flow model

The Sobek open channel flow model simulates water flow in rivers by solving the Saint-Vanant equations with the so-called staggered grid numerical scheme (Stelling & Duinmeijer, 2003).

The Sobek schematization “Elbe at Magdeburg” is shown in Figure 5. The water system model network has the following characteristics:

- one branch in the south
- one branch in the north
- two branches in the centre, one representing the main river and one represents the Old Elbe branch
- cross sections
- observation points
- one weir to close the Old Elbe branch upstream.

The upstream boundary condition is a discharge time series, as downstream boundary condition a rating curve (discharge-water level relation) is set.

Task:

- Open the Sobek model.
- Inspect the network: find the observation points and the structure.
- Look at the inflow boundary.
- Run the model.
- Inspect the side-view for the two routes “Elbe” and “Old Elbe”.
- Look at the hydrographs for the two observation points.

For modelling with Sobek see the user manual (Deltares, 2014).

4.4 The RTC-Tools real-time control model

The RTC-Tools model (Deltares, 2013) addresses the control of the weir which is represented in the Sobek model as structure node. The control is based on water level observations at the Schönebeck gauge in the upstream part of the model. The gauge is represented in the Sobek model as observation point. The control flow is given in Figure 6 as a decision tree.

A trigger evaluates if the observed water level at Schönebeck is greater than 54 m. If the condition is true, the weir is opened, if not, the weir is closed. This simple operational protocol ensures sufficient water depth for cargo ship navigation in the main channel of the Elbe during normal condition and reduces the water level during high water conditions.

Task:

- Open the file `rtcToolsConfig.xml`.
- Find the trigger and rule elements from the flow chart in Figure 6.

See the manual (Deltares, 2012) for details on working with RTC-Tools.

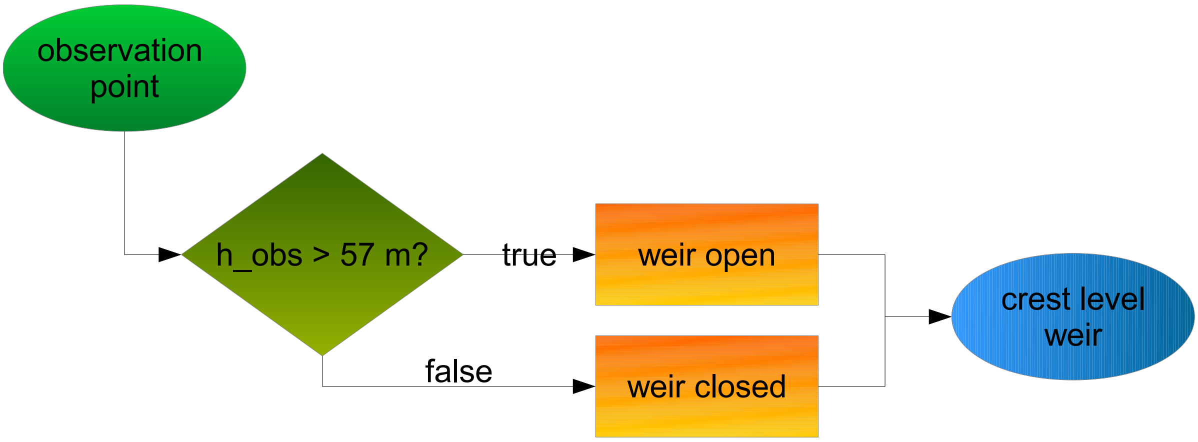


Figure 6: Control flowchart modelled with RTC-Tools

4.5 Coupling with OpenMI

The processes interactions of human control and open channel flow is as follows:

- the crest level of the weir is controlled in dependence of the current water level at the observation point and
- the control of the weir has an impact on the water system:
  - if the weir is open, water can flow through the main branch and the Old Elbe branch
  - if the weir is closed, the water flows through the main Elbe branch only.

To model this interaction, bi-directional data exchange has to be configured as follows:

- Sobek provides the water level at Schönebeck gauge to RTC-Tools
- RTC-Tools provides the crest level for the weir to Sobek.

Task:

- Open the OpenMI configuration editor.
- Load the Sobek model into the OpenMI configuration editor.
- Load the RTC-Tools model into the OpenMI configuration editor.
- Add a trigger component to the composition. Note that the OpenMI trigger should not be confused with the RTC-Tools trigger element.
- Add a connection from the RTC-Tools model to the Sobek model and configure the connection as shown in Fig. 9.
- Add a connection from the Sobek model to the RTC-Tools model and configure the connection as shown in Fig. 8.
- Add a connection from the Sobek model to the OpenMI trigger and configure the connection as shown in Fig. 10. Choose an arbitrary exchange item from the Sobek model.

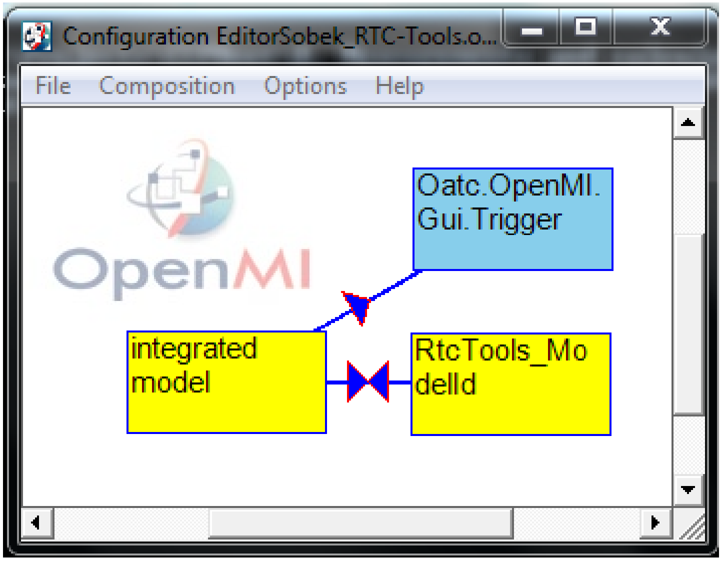


Figure 7: OpenMI Configuration Editor with a Sobek- an RTC-Tools- and an OpenMI trigger component

– Save the composition.

The OpenMI composition should look like the one in Figure 11.

The functional principle of the data exchange is shown in Fig. 11 can be summarized as follows (Becker & Talsma, 2014):

– The model component that asks first computes last.

– The model that asks gives the guess (i. e. data from the previous time step).

So for a given time step RTC-Tools computes before Sobek, so RTC-Tools uses Sobek data from the previous time step.

4.6 Coupled simulation and simulation results

Task:

– Run the OpenMI composition via the Run properties window (Fig. 12).

– Open the Sobek model that has been running within the OpenMI coupled simulation.

– Inspect the side views for the routes “Elbe” and “Old Elbe”. For the latter one, add the coverage “Crest level (s)”.

– Inspect the hydrographs of the two observation points and the crest level.

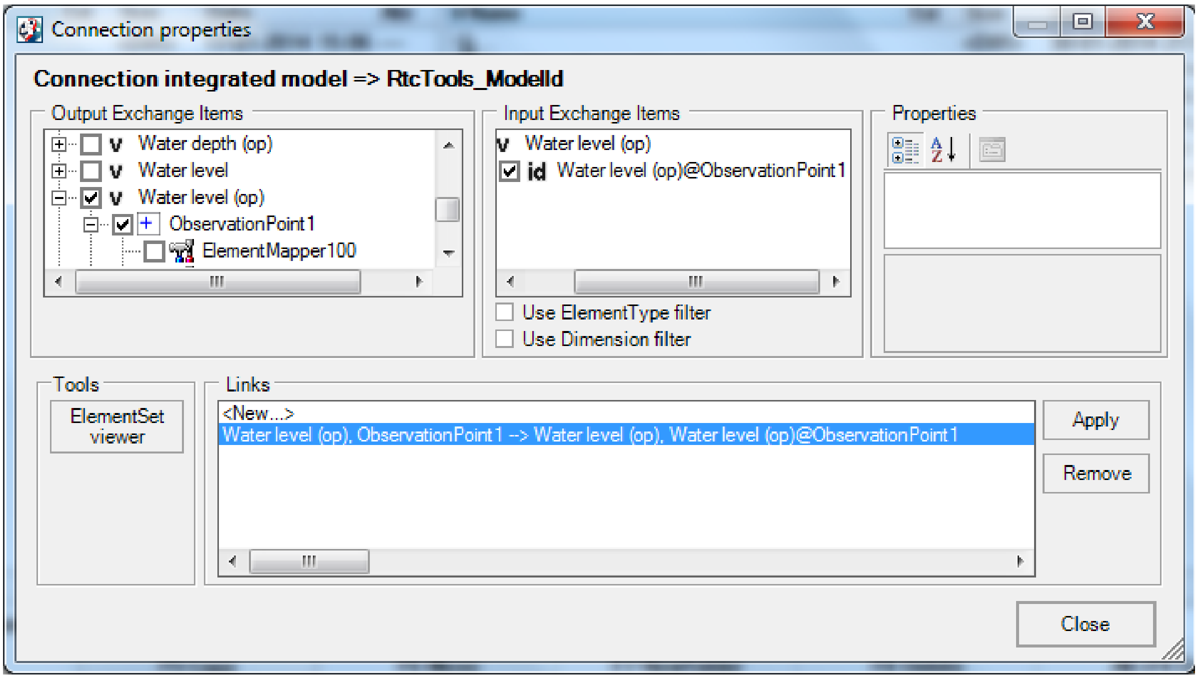


Figure 8: Connection properties Sobek– RTC-Tools

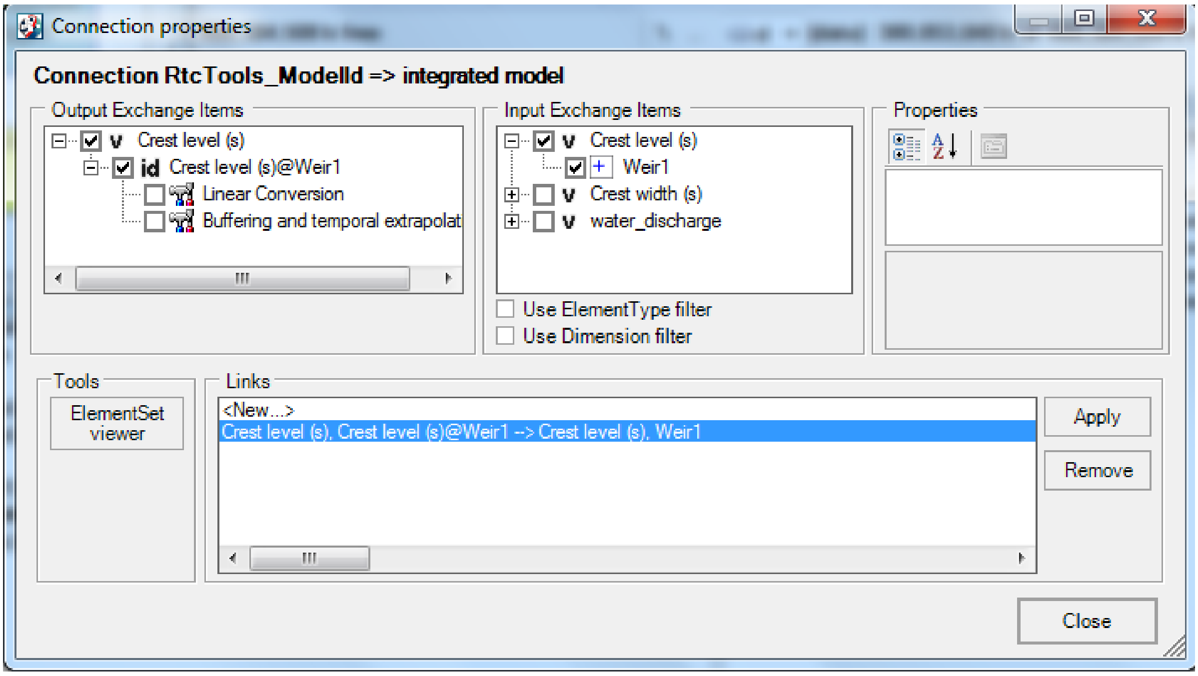


Figure 9: Connection properties RTC-Tools– Sobek

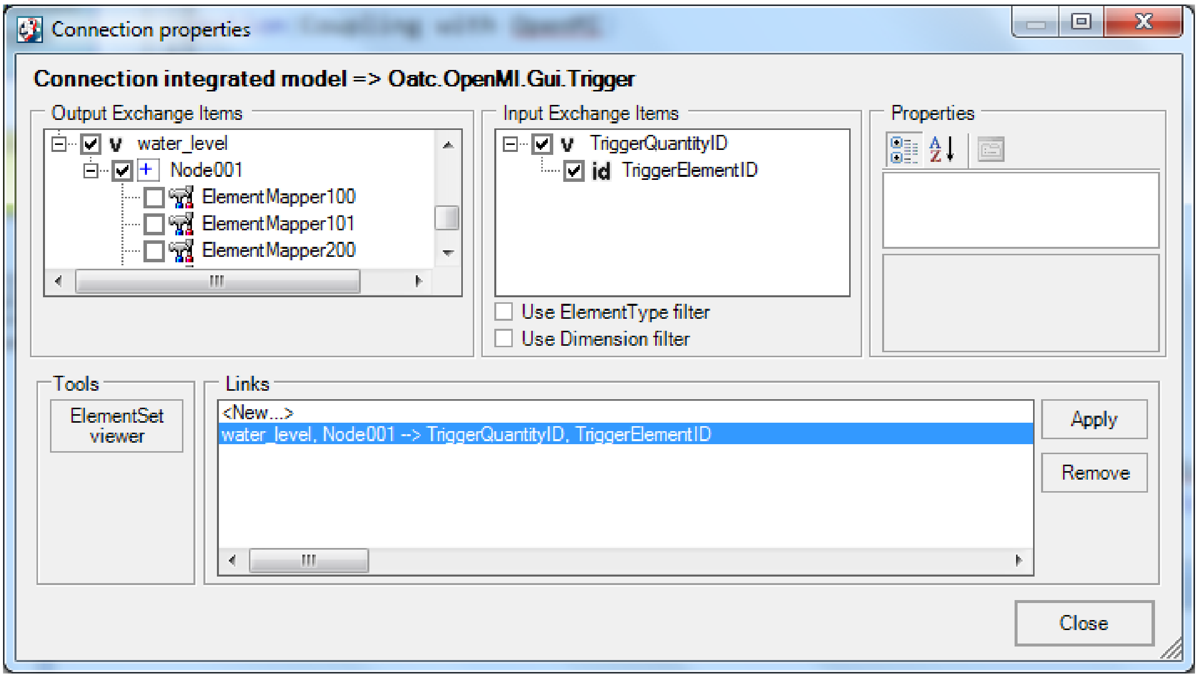


Figure 10: Connection properties Sobek– OpenMI trigger

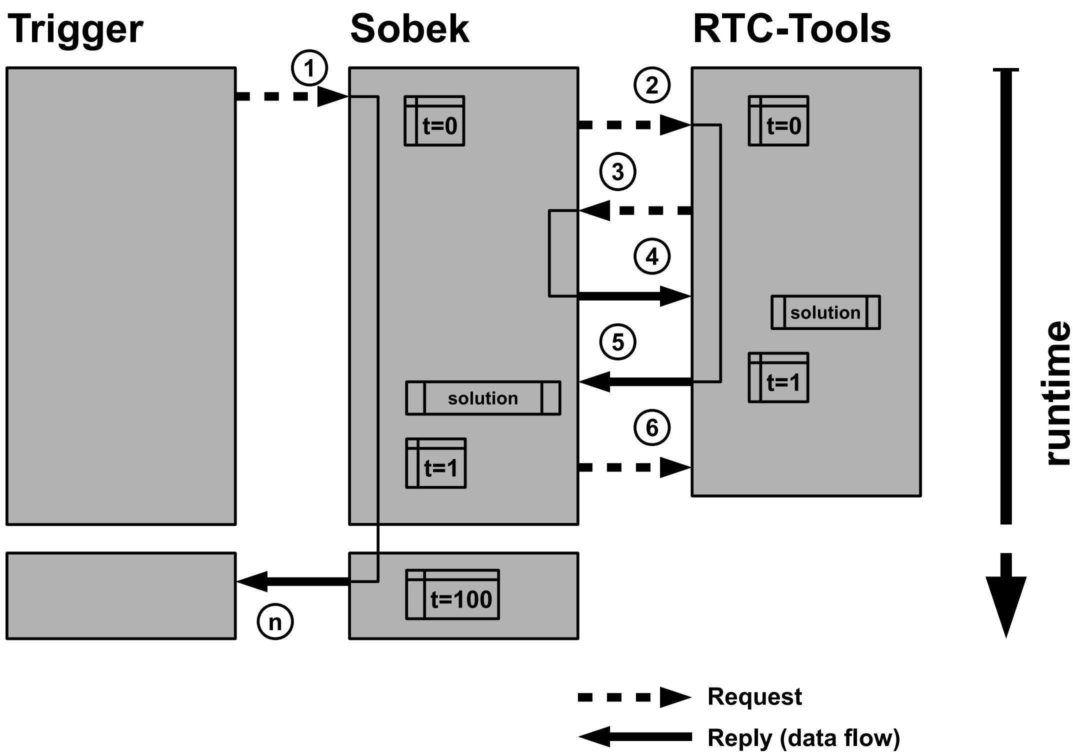


Figure 11: Request-reply mechanism for an OpenMI composition with Sobek and RTC-Tools.

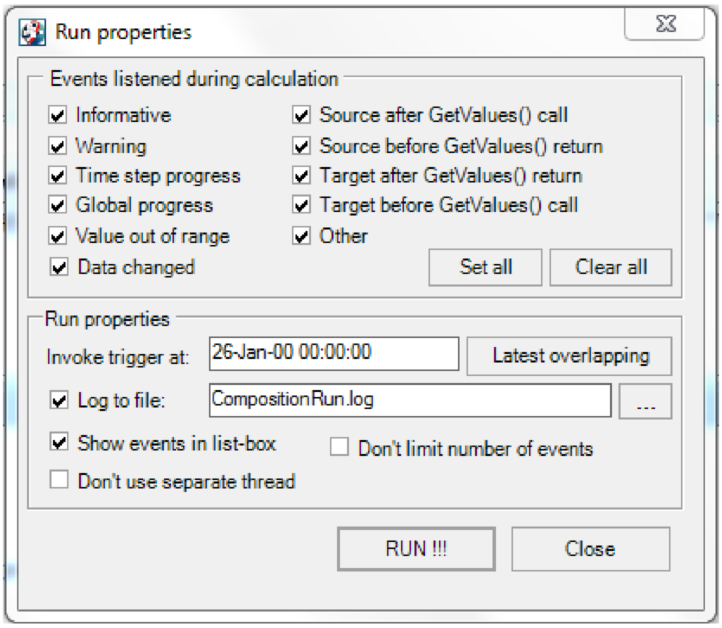


Figure 12: OpenMI configuration editor Run properties window

Fig. 13 shows simulation results from the Sobek model with an uncontrolled weir (Fig. 13a) and the simulation results from the coupled simulation Sobek– RTC-Tools (Fig. 13b), where the weir is controlled in dependence of the water level at Schönebeck gauge (observation point 1). In the coupled simulation the water level at the observation point "Magdeburg" (observation point 2) remains below the flood warning level of 54.8 m, because the weir has been opened after the water level at Schönebeck gauge has reached 57 m. At the bifurcation point the water divided into the Old Elbe branch which results in a lower water level in the main branch of the Elbe.

References

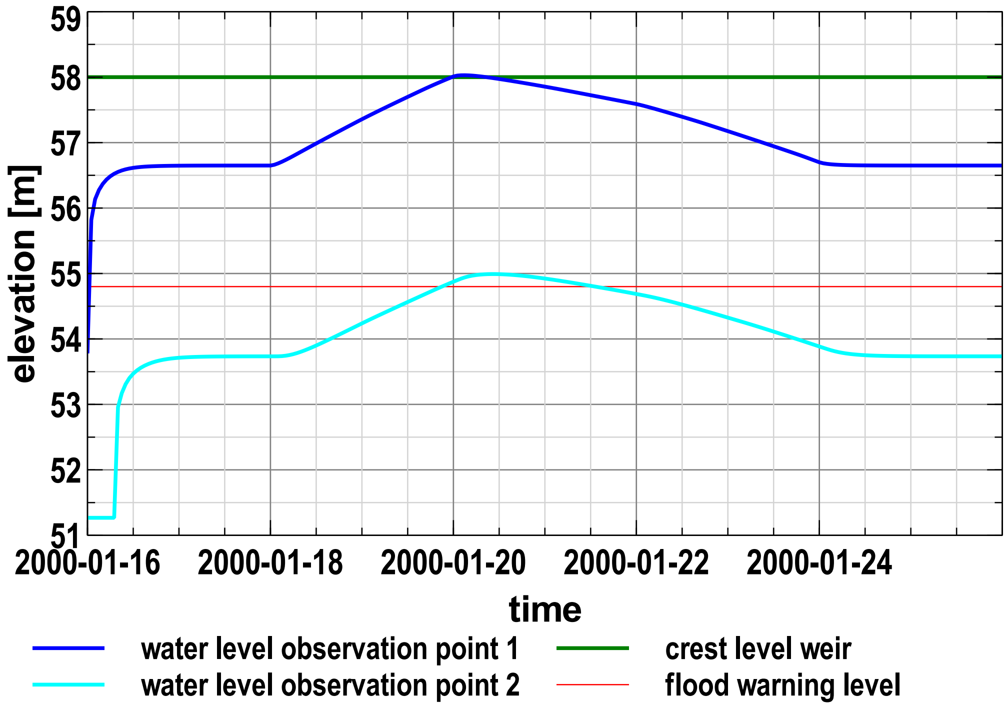
Becker, B. : Inzet RTC-Tools voor het boezemmodel "Wetterskip Fryslân". Report 1205773-000, Deltares, Delft (2013). In Dutch.

Becker, B. , Dahm, R. , van Heeringen, K.-J. , Goorden, N. , Kramer, N. , Kooij, K. , Gooijer, J. , Jansen, J. : Op zoek naar een optimaal ontwerp voor een groot uitwateringsgemaal in het Lauwersmeer. H2O, Vol. 44 (13), pp. 11–13 (2012a).

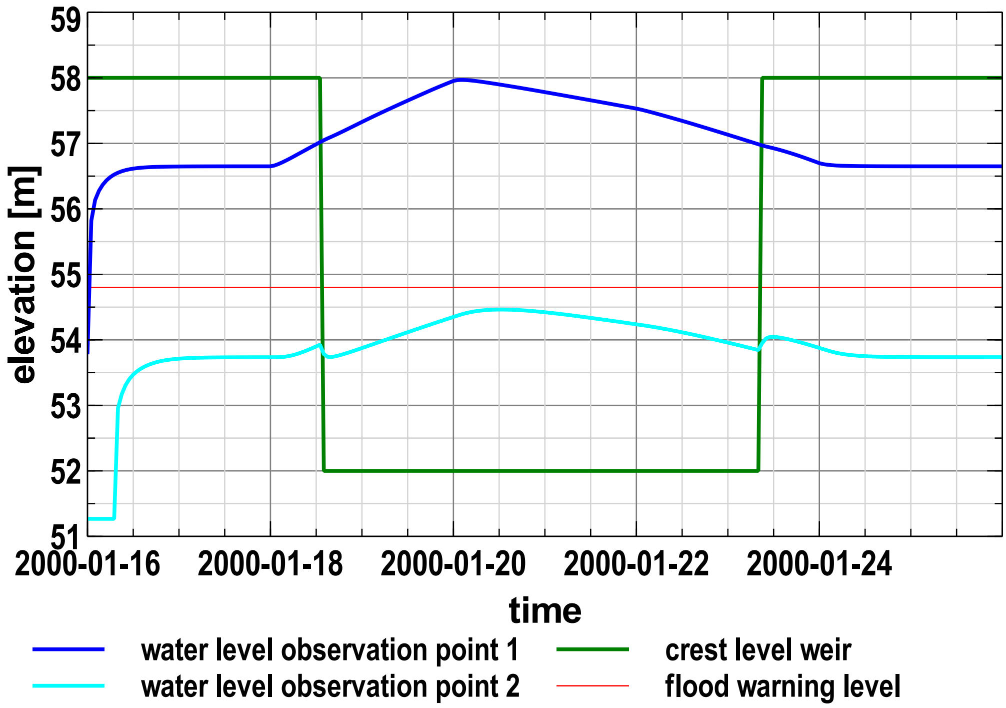
Becker, B. , Gao, Q. : Koppelen Sobek-modellen "Wetterskip Fryslân" en "Waterschap Noorderzijlvest" via OpenMI. Report 1204514-000-ZWS-0007, Deltares, Delft (2012). In Dutch.

Becker, B. , Talsma, J. : On the external and iterative coupling of multiple open channel flow models with OpenMI. Revista de Ingeniería Innova, Vol. accepted for publication (2014).

Becker, B. P. , Schüttrumpf, H. : An OpenMI module for the groundwater flow simulation programme Feflow. Journal of Hydroinformatics, Vol. 13 (1), pp. 1–13 (2010), ISSN 1464-7141, doi:10.2166/hydro.2010.039. URL `http://www.iwaponline.com/jh/013/jh0130001.htm`.



(a) simulation results from the Sobek model (channel flow)



(b) simulation results from the coupled flow simulation with Sobek (channel flow) and RTC-Tools (control)

Figure 13: Simulation results

Becker, B. P. J. : Zur gekoppelten numerischen Modellierung von unterirdischem Hochwasser. Dissertation, RWTH Aachen, Fakutlät für Bauingenieurwesen, Aachen (2010). In German, URL http://d-nb.info/100850209X/34.

Becker, B. P. J. , Forberig, S. , Flögel, R. , Schüttrumpf, H. , Köngeter, J. : On the Determination of Groundwater Levels for Hazard Maps of Groundwater Head Rise Induced by High Water. Wasserwirtschaft, (12), pp. 10–16 (2011), ISSN 0043-0978.

Becker, B. P. J. , Schwanenberg, D. , Schruff, T. , Hatz, M. : Conjunctive real-time control and hydrodynamic modelling in application to Rhine River. Proceedings of 10th International Conference on Hydroinformatics, TuTech Verlag TuTech Innovation GmbH, Hamburg, Germany (2012b), ISBN 978-3-941492-45-5.

Becker, B. P. J. , Talsma, J. , Gao, Q. , Ruijgh, E. : Coupling of multiple channel flow models with OpenMI. Proceedings of 10th International Conference on Hydroinformatics, Hamburg, Germany (2012c), ISBN 978-3-941492-45-5.

Deltares: RTC-Tools a software package for modelling real-time control / Technical Reference Manual and Configuration Guidelines. Deltares, Delft (2012). Version: 0.1.0.22313.

Deltares: RTC-Tools a toolbox for real-time control of hydraulic structures (2013). Published: Deltares, URL http://oss.deltares.nl/web/rtc-tools.

Deltares: SOBEK 3 / D-Flow 1D and D-Real Time Control in Delta Shell / User Manual. Deltares, Delft (2014). Version: 3.2.1.31793.

Fairbanks, J. , Panday, S. , Huyakorn, P. S. : Comparisons of linked and fully coupled approaches to simulating conjunctive surface/subsurface flow and their interactions. MODFLOW 2001 and other Modeling Odysseys - Conference Proceedings, pp. 356–361, Golden, CO (2001).

Gregersen, J. , Gijsbers, P. , Westen, S. : OpenMI: Open modelling Interface. Journal of Hydroinformatics, Vol. 9 (3), pp. 175–191 (2007), doi:10.2166/hydro.2007.023.

Huang, G. , Yeh, G.-T. : An integrated media, integrated processes watershed model – WASH123D: part 3 – a comparative study on different surface water/groundwater coupling approaches. Binning, P. J. , Engesgaard, P. K. , Dahle, H. K. , Pinder, G. F. , Gray, W. G. (eds.) Proceedings of the XVI International Conference on Computational Methods in Water Resources, Copenhagen, Denmark (2006). URL http://proceedings.cmwr-xvi.org/contributionDisplay.py?contribId=88&sessionId=10&confId=a051.

Konikow, L. F. , Bredehoeft, J. D. : Ground-water models cannot be validated. Advances in Water Resources, Vol. 15, pp. 75–83 (1992).

Moore, R. , Gijsbers, P. , Fortune, D. , Gregersen, J. , Blind, M. : OpenMI Document Series: Part A – Scope for the OpenMI. Version 1.0 edn. (2005). URL http://www.openmi.org/openminew/documents/A_OpenMI_Scope.pdf.

Moore, R. V. , Tindall, C. I. : An overview of the open modelling interface and environment (the OpenMI). Environmental Science & Policy, Vol. 8 (3), pp. 279–286 (2005), doi:10.1016/j.envsci.2005.03.009.

Morita, M. , Yen, B. : Modeling of Conjunctive Two-Dimensional Surface-Three-Dimensional Subsurface Flows. Journal of Hydraulic Engineering, Vol. 128 (2), pp. 184–200 (2002), doi:10.1061/(ASCE)0733-9429(2002)128:2(184).

Morita, M. , Yen, B. C. : Numerical methods for conjunctive two-dimensional surface and three-dimensional sub-surface flows. International Journal for Numerical Methods in Fluids, Vol. 32, pp. 921–957 (2000).

Schellekens, J. , Becker, B. P. J. , Donchyts, G. , Goorden, N. , Hoogewoud, J. C. , Patzke, S. , Schwanenberg, D. : OpenStreams: Open Source Components as Building Blocks for Integrated Hydrological Models. Geophysical Research Abstracts, Vol. 14 EGU2012, p. 3953, Vienna, Austria (2012). URL http://adsabs.harvard.edu/abs/2012EGUGA..14.3953S.

Stelling, G. S. , Duinmeijer, S. P. A. : A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. International Journal Numerical Methods In Fluids, Vol. 43, pp. 1329–1354 (2003).

Werner, M. , Schellekens, J. , Gijsbers, P. , van Dijk, M. , van den Akker, O. , Heynert, K. : The Delft-FEWS flow forecasting system. Environmental Modelling & Software, Vol. 40, pp. 65–77 (2013), ISSN 13648152, doi: 10.1016/j.envsoft.2012.07.010. URL http://linkinghub.elsevier.com/retrieve/pii/S1364815212002083.

Feedback on this lecture notes

Comments and feedback on this document are welcome, please send them via e-mail to Bernhard Becker (bernhard.becker@deltares.nl, subject: feedback on OpenMI lecture notes).

Appendix B: OpenMI Slides

CIPRNet

Critical Infrastructure Preparedness and Resilience Research Network
www.ciprnet.eu

Introduction in OpenMI

Andreas Burzel and Bernhard Becker (Deltares)
andreas.burzel@deltares.nl | bernhard.becker@deltares.nl

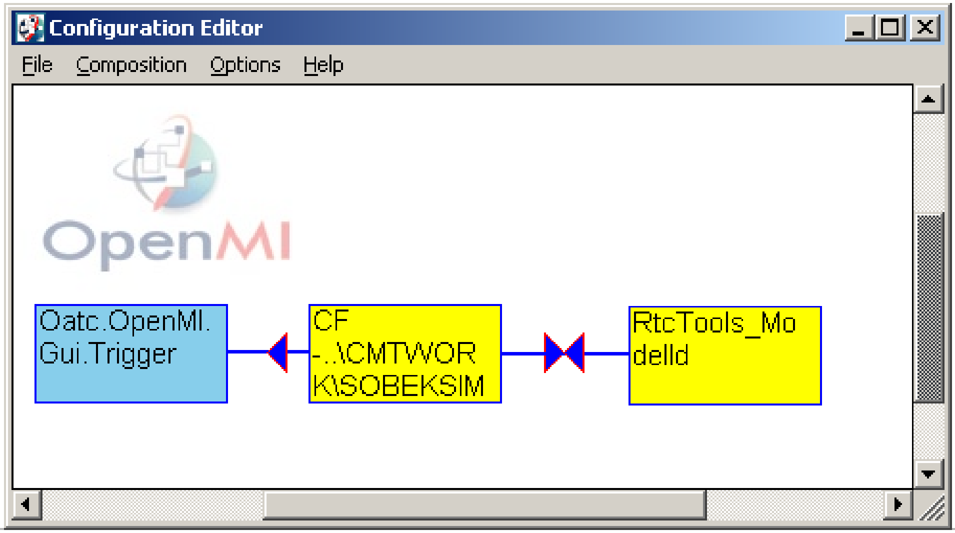
Modelling, Simulation and Analysis of Critical Infrastructure Training School (Edition 0)

Deltares Headquarters – Delft (The Netherlands) – 3-4 February 2014

What is OpenMI?

OpenMI is an open model interface standard for hydro-related models developed by the OpenMI Association

- Designed for water-related models
- For legacy code and new code
- Data-exchange during runtime per time step
- Open source
- Used already by several institutions (Deltares, DHI, BAW, RWTH Aachen University, Université de Liège, US Geological Survey, ...)



example:
RTC-Tools and Sobek in the OpenMI configuration editor

OpenMI history

HarmonIT - OpenMI v1.0

- OpenMI was developed by 14 organizations from 7 countries in the EU-project HarmonIT in order to facilitate the simulation of interacting processes, particularly environmental processes
- the first version has been released as the OpenMI Standard v1.0 (.Net version)

OpenMI-Life - OpenMI v1.4

- Further development has been performed in the OpenMI-Life project with a consortium of 10 partners from 5 countries
- release of v1.4 (.Net, Java), foundation of the OpenMI Association

Released - OpenMI v2.0

- Several new features are introduced, including a more flexible way of linking, more flexibility in the overall control flow, less difference between temporal and spatial models
- A new user interface (GUI) and a software development kit (SDK) allow users to make their models 2.0 compliant


When to apply OpenMI?

Coupling of models of different processes

- one model for each process
- both processes are of similar relevance
- processes on different time scales

Coupling of models of the same type

- models belong to different institutions
- models are used coupled and uncoupled (maintenance, calibration, local studies)

A quick comparison

- OpenMI: data exchange during runtime per time step
- Delft-FEWS: data exchange after run for a simulation period

Different Types of Coupling

Simultaneous coupling

- the highest level of model coupling
- different processes, including their interactions, are represented in one equation system

Iterative coupling

- exchange data between models during runtime in two directions and iterate the exchange of data until a certain convergence criterion is achieved

External coupling

- data exchange per time step, successively, but without iterations.

→ OpenMI supports iterative coupling and external coupling.

Who should apply OpenMI?

“The long term aim is that the OpenMI should become the European and global standard for model linking in the environmental domain.” (from the OpenMI-life website)

Researchers that develop source code for their studies

- research code can be coupled with OpenMI compliant models

Developers of integrated (hydrological) modelling tools

- coupling of surface/subsurface flood models

Consultants that need dedicated model coupling

- flexible, standardized coupling technique
- use the OpenMI standard for more than one coupling task

Multidisciplinary studies

- CIPRNet - coupling of CI models

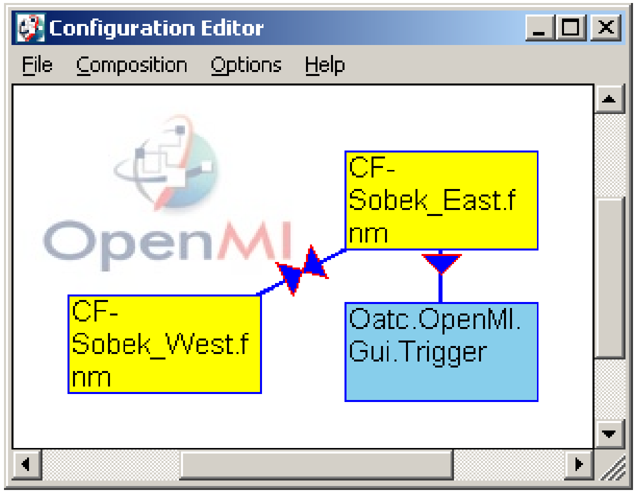
Setting up an OpenMI-Composition

omi-file: the OpenMI-Compliant Component

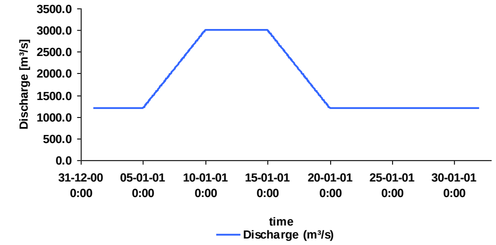
- Where is the DLL with the computational core and OpenMI-Interface?
- Where are the input files?
- What else? (Command line arguments)

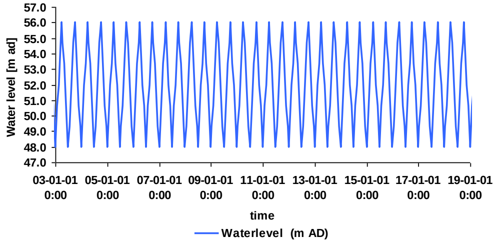
opr-file: the OpenMI-Composition

- Which components (i. e. models)?
- How coupled?
- Which simulation period?
- Where is the Trigger linked with?

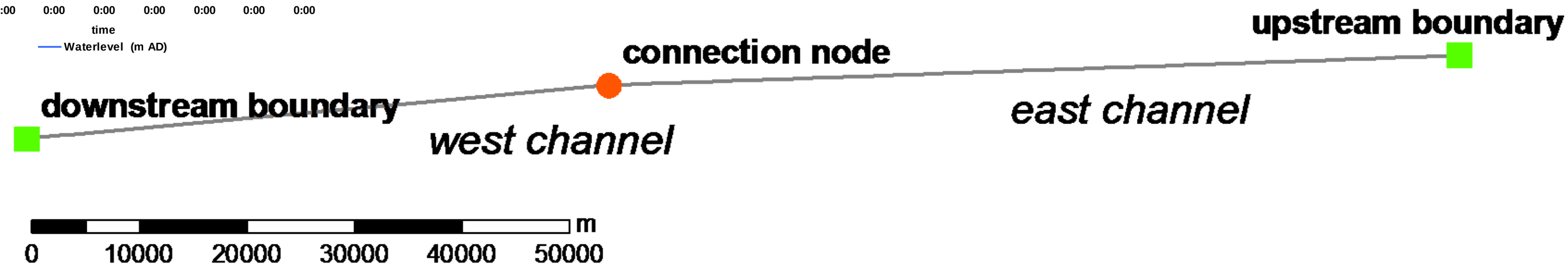


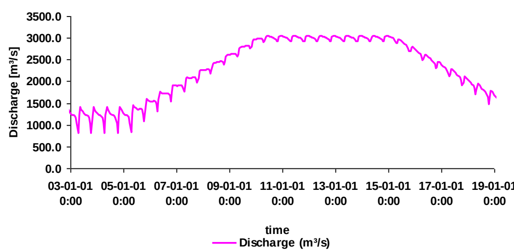
Example case: Sobek-West ←→ Sobek East

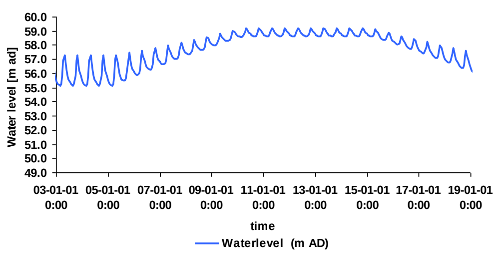




sub-model "channel west"			sub-model "channel east"	
input	discharge	←	output	discharge
output	water level	→	input	water level

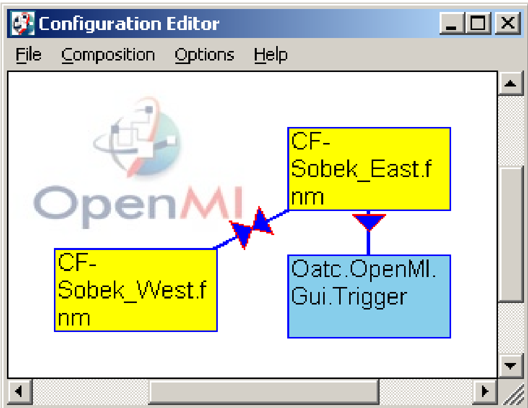




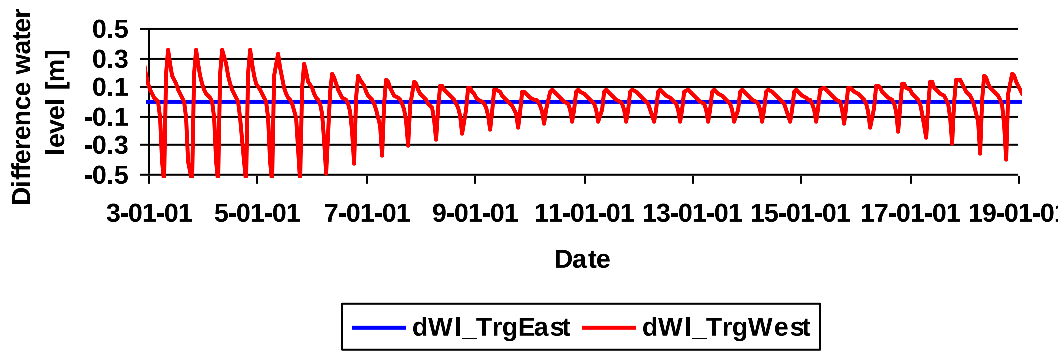


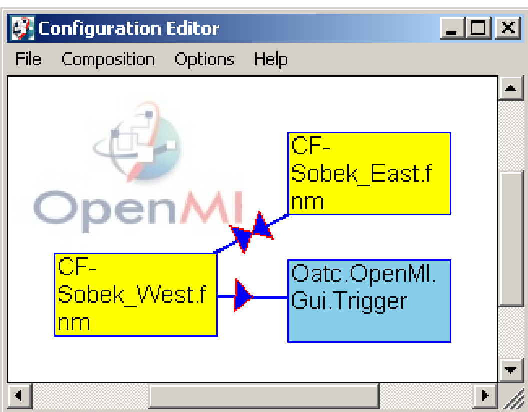
Exchange of results and boundary conditions between multiple models

External coupling (time-lagged coupling)

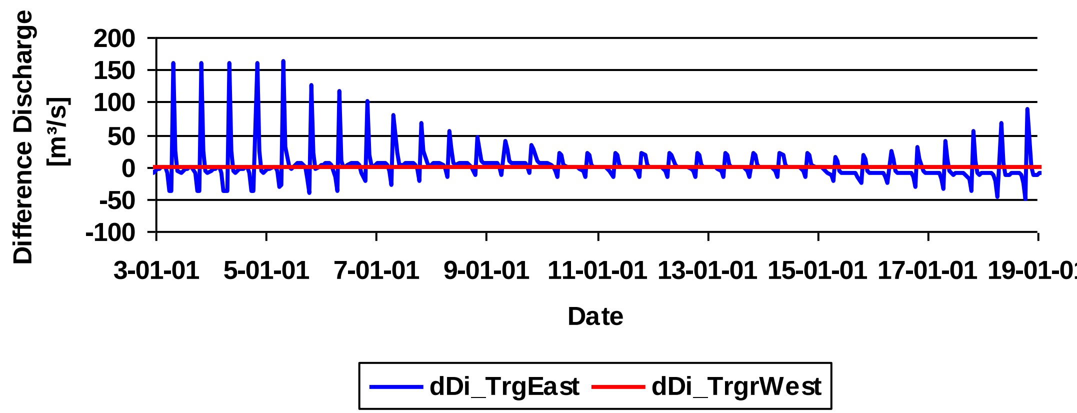


mass conservation





momentum conservation

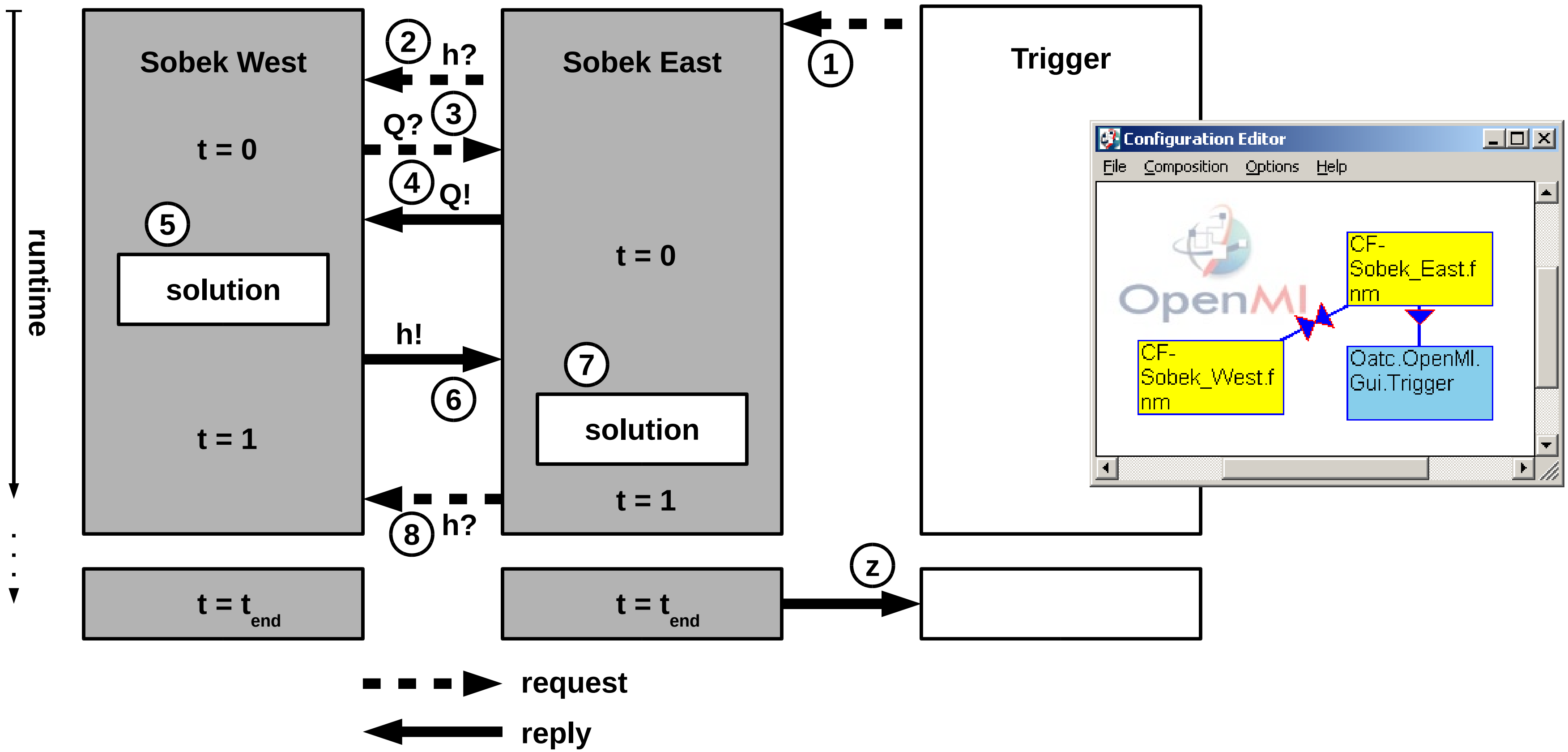


Position of the trigger impacts the result

external coupling has a time lag

Deltares

Request-reply-mechanism



The model component that asks first computes last.

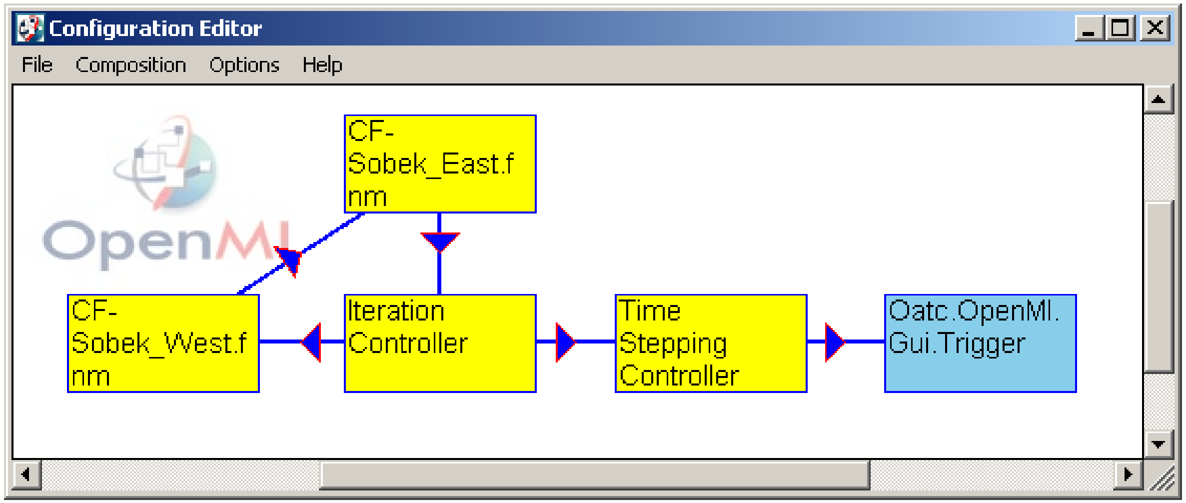
The model that asks gives the guess.

The quantity which is computed first has the same value in both of the models.

Deltares

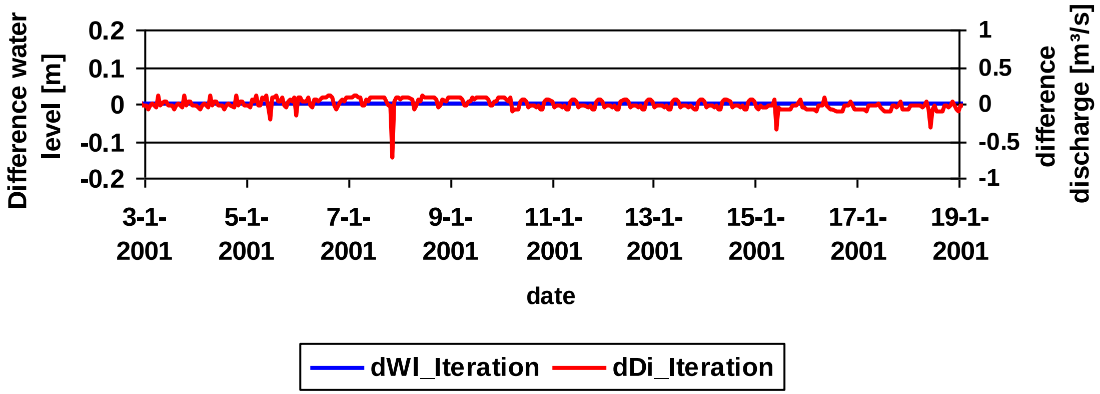
Iterative coupling

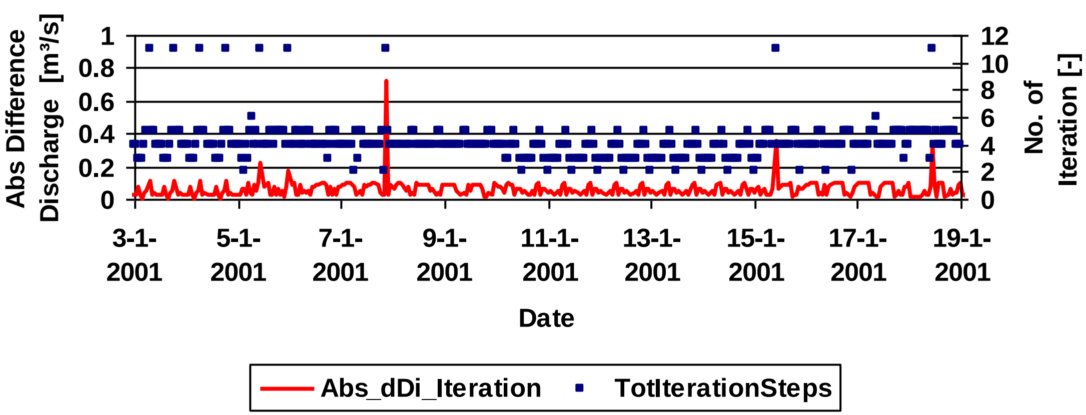
OpenMI compliant component “iteration controller”



objective: reduce difference in discharge by iteration
better momentum conservation
maximum 12 iterations per time step or dQ < epsilon = 0.01

Iterative coupling

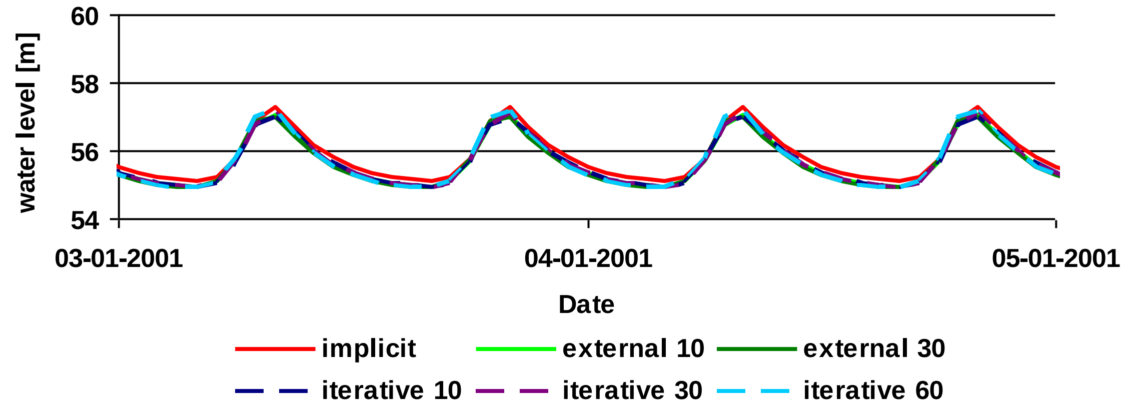


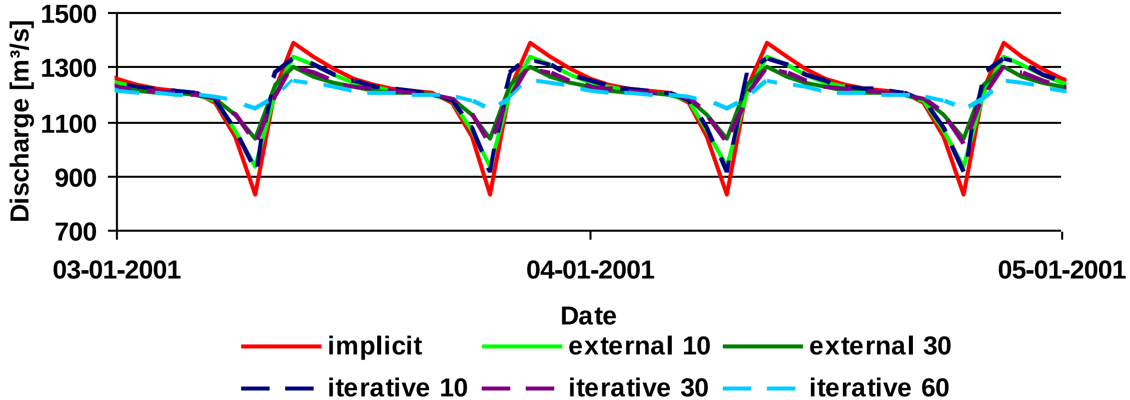


Iteraties increase accuracy by repeating data exchange per time step

Deltares

Iterative and external coupling: test case results





coupling method (min)	computing time
ternally coupled (10)	30 s
externally coupled (10)	27 s
externally coupled (30)	14 s
externally coupled (30)	12 s
iteratively coupled (10)	192 s
iteratively coupled (30)	50 s
iteratively coupled (60)	43 s
implicit solution	3 s

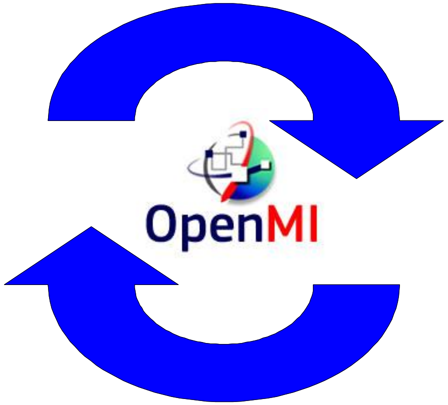
External coupling as accurate as iterative coupling, but more controllable and less computational expensive

Example: SOBEK ←→ WANDA

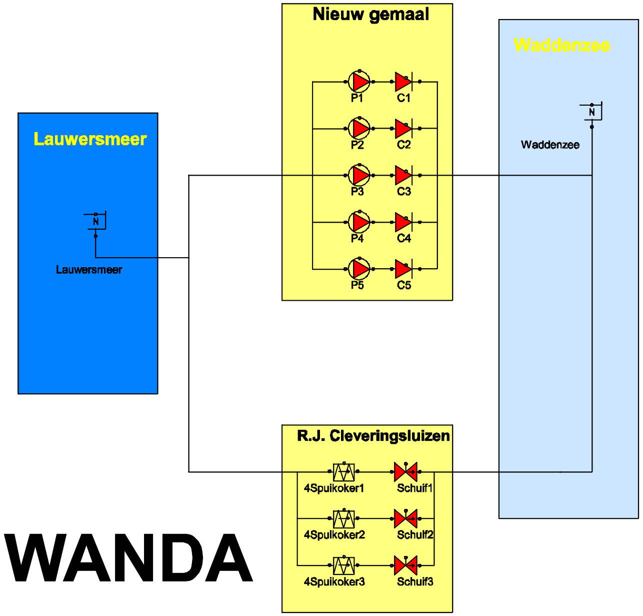
Design of a pump station for lake Lauwersmeer (the Netherlands)

- more extreme rainfall events and rising sea level expected
- drainage of polder areas must be facilitated with a pump station

open channel flow

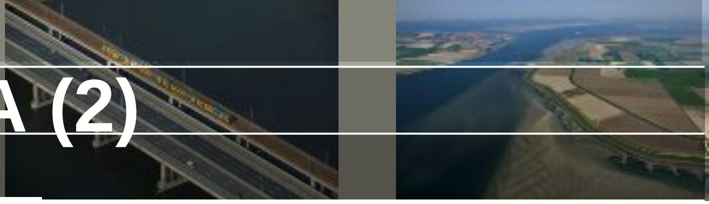


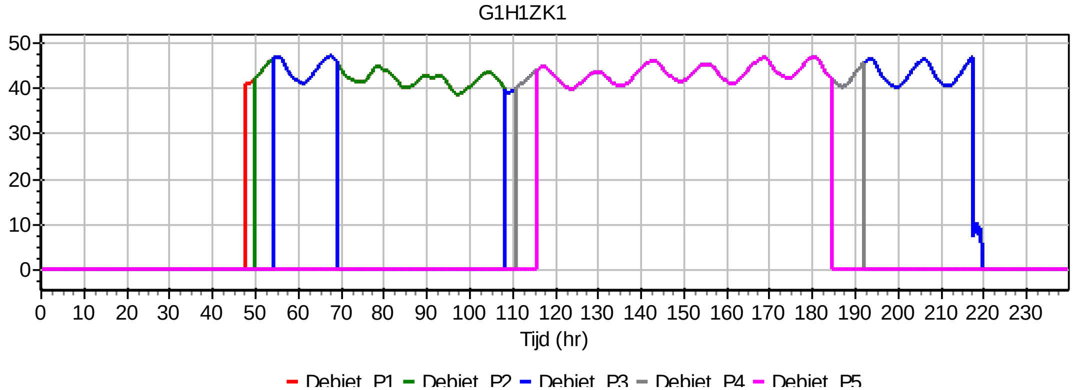
pump and sluice hydraulics



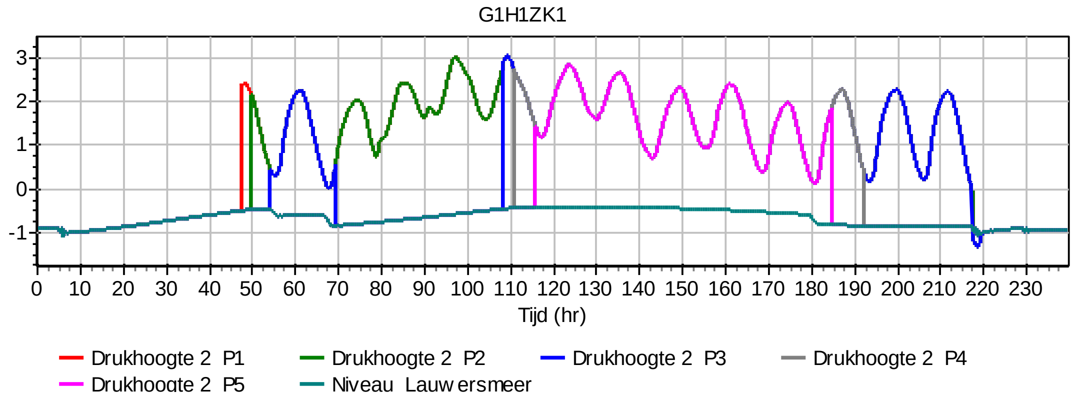
Deltares

Example: SOBEK ←→ WANDA (2)

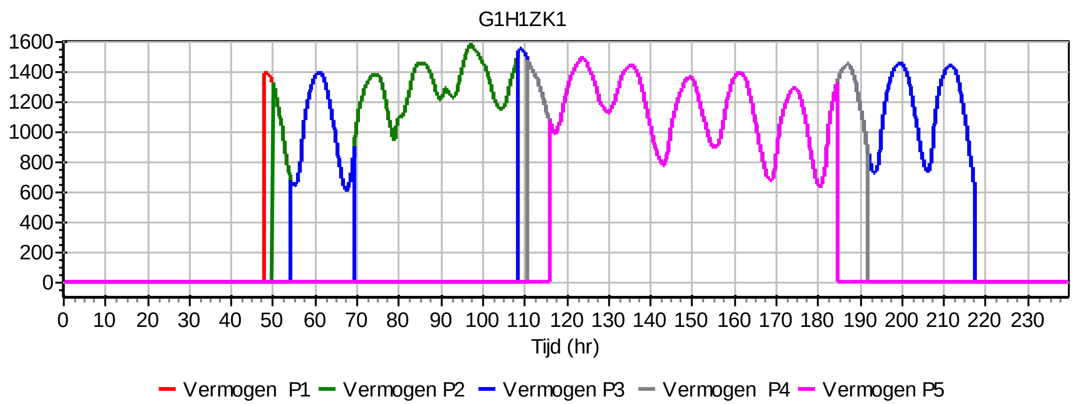




discharge from WANDA for SOBEK



Pressure head from SOBEK (tidal influenced)



power consumption from WANDA for design

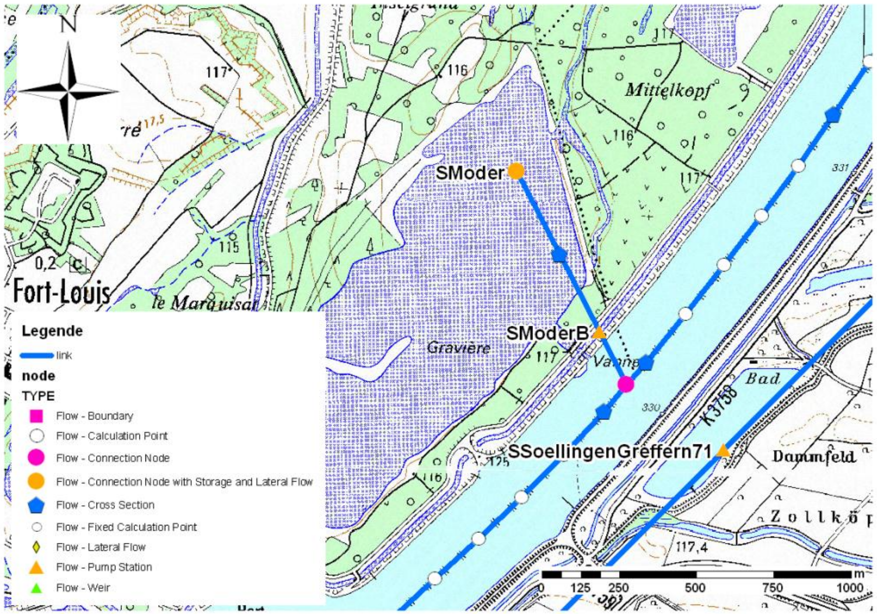
Deltares

Example: SOBEK ←→ RTC-Tools

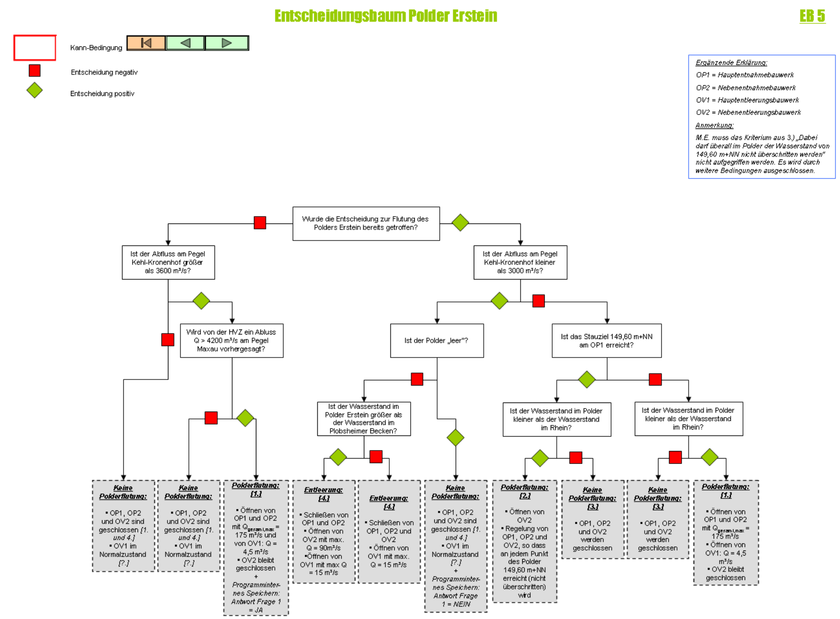
Control of the Oberrhein (upper Rhine) water system:

Decision tree and open channel system

SOBEK: open channel flow



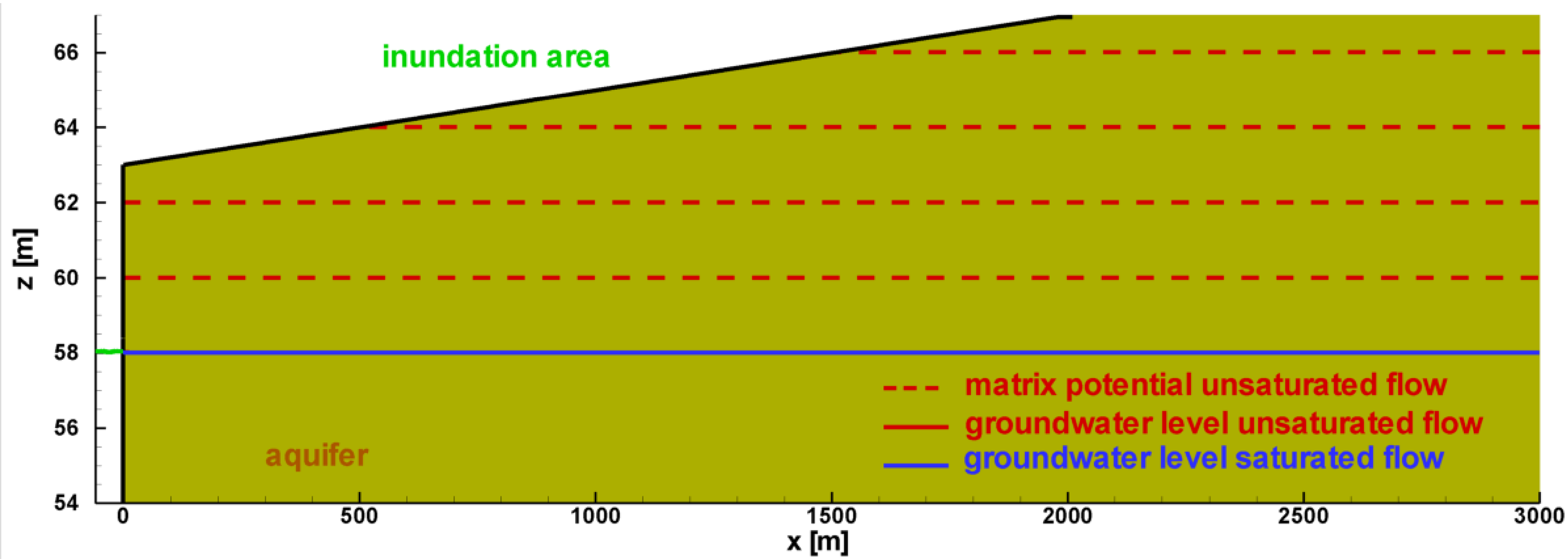
RTC-Tools: human operations (control)



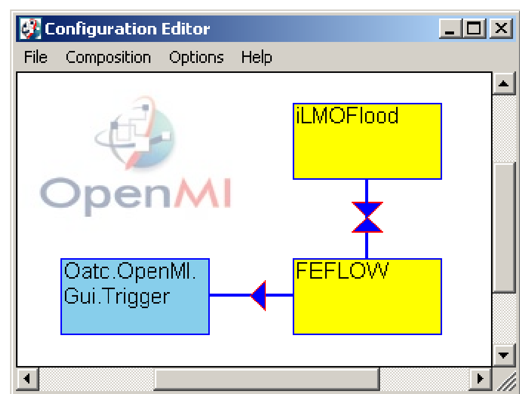
Sobek		RTC-Tools
Channel flow (Q, h)	→	**Water system state**
Control parameter	←	**Control parameter (crest level, turbine discharge)**

Deltares

Example: Ilmoflood ←→ Feflow



Simulation period: 14 days



Two coupled simulations:

Ilmoflood ←→ Feflow saturated

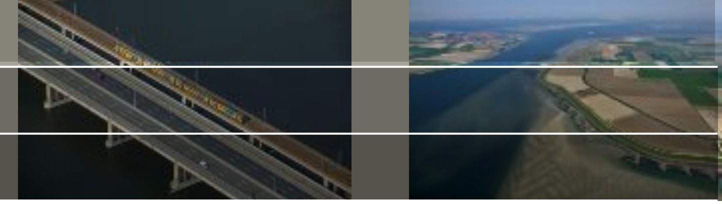
Ilmoflood ←→ Feflow unsaturated

coupling:

Ilmoflood		Feflow
head	**→**	**leakage**
flow	**←**	**flow**

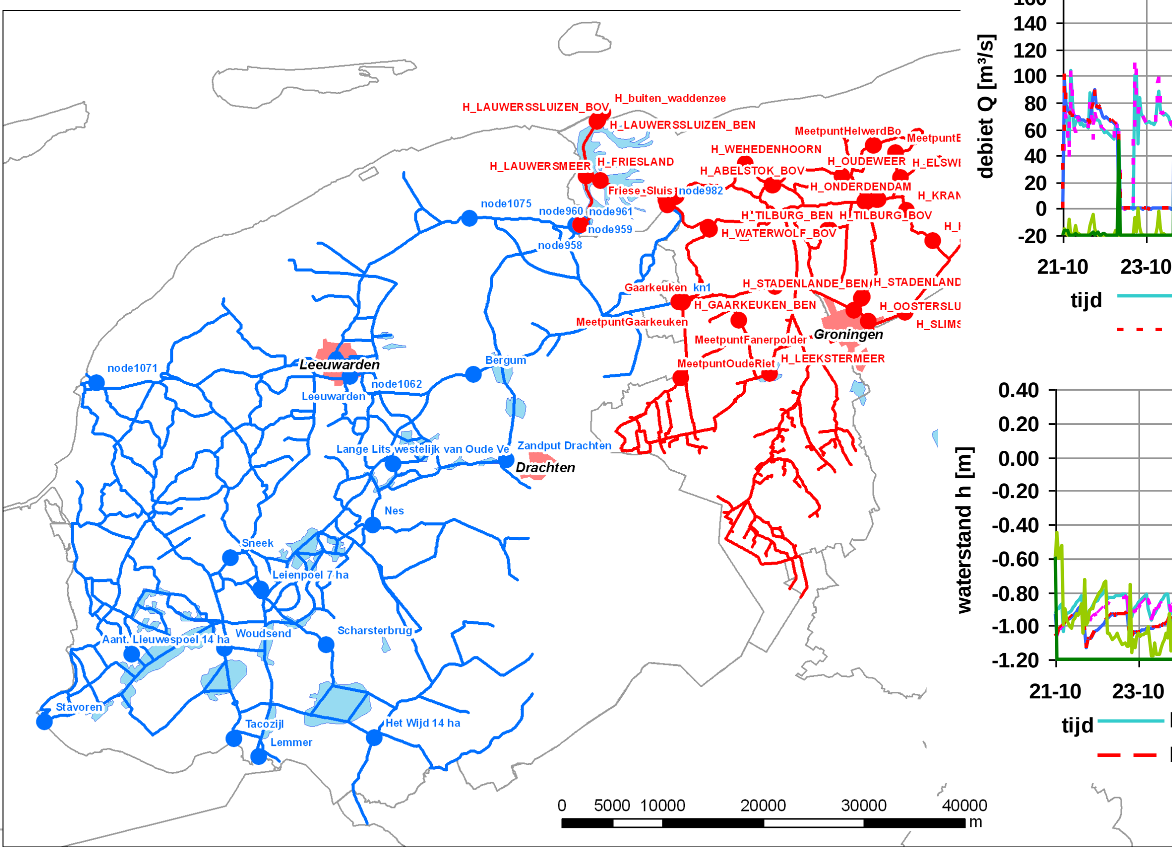
Deltares

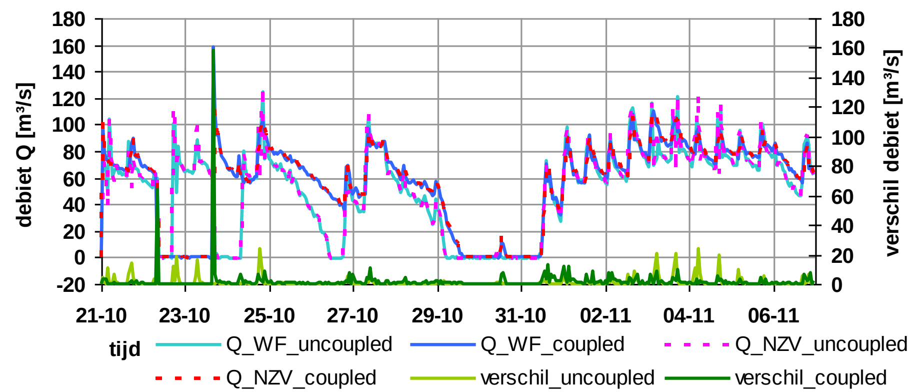
Example: Boezemmodellen

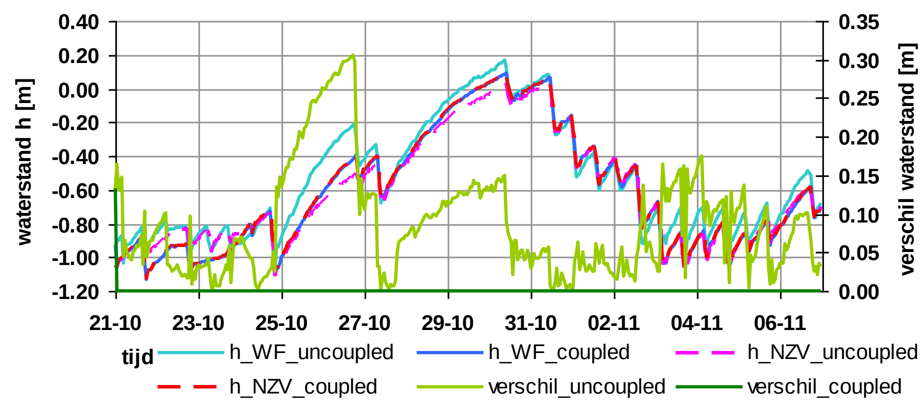


Boezemmodellen Wetterskip Fryslân and Noorderzijlvest coupled at three connection points

One water system, two water authorities

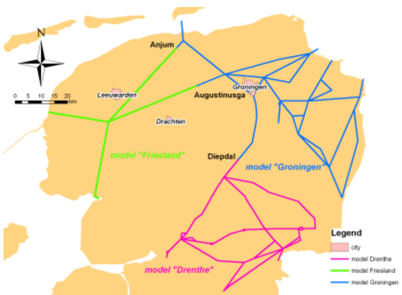


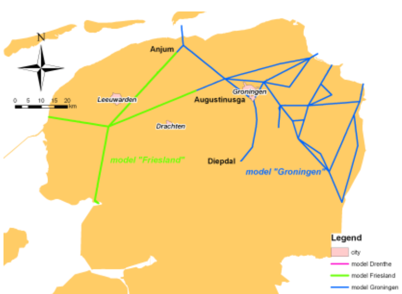


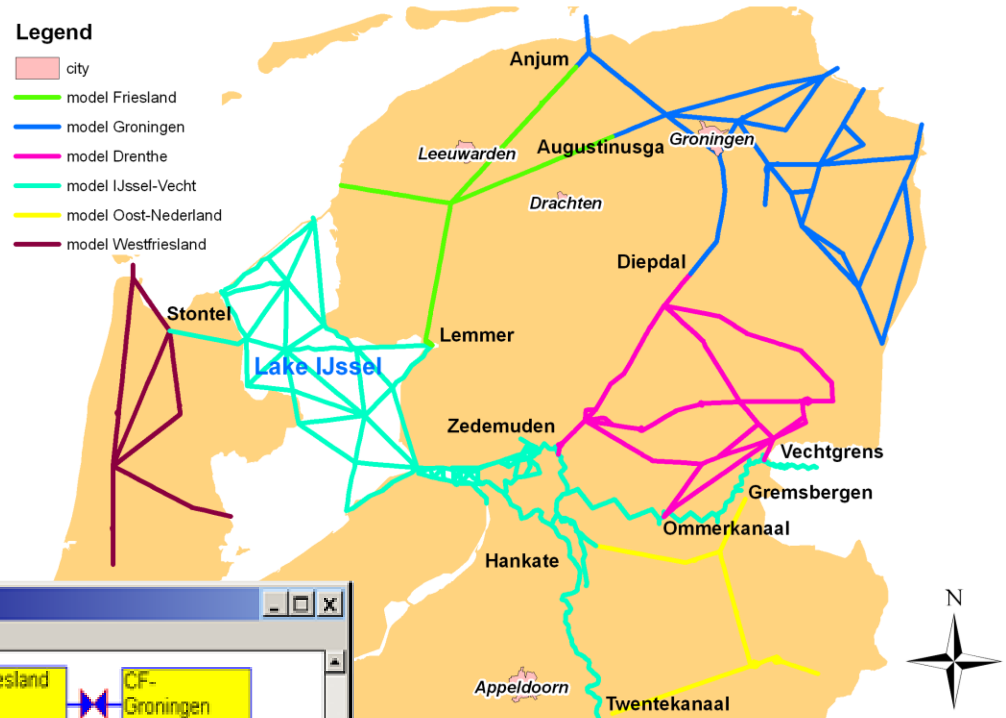


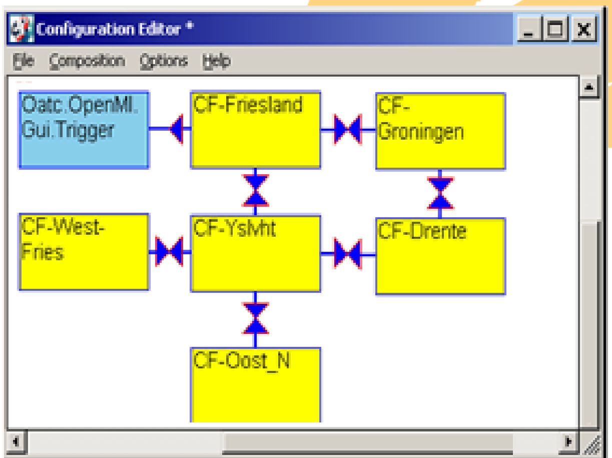
Dutch Large Scale SOBEK model

From 2 to 6 model coupled:



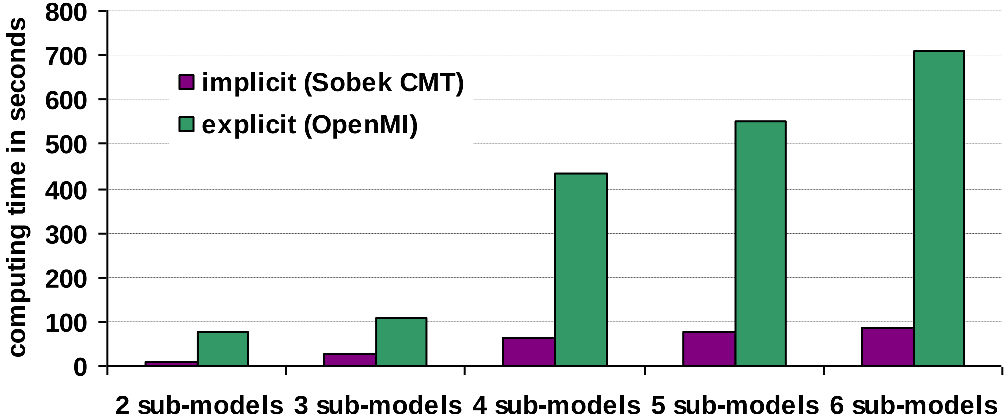






Deltares

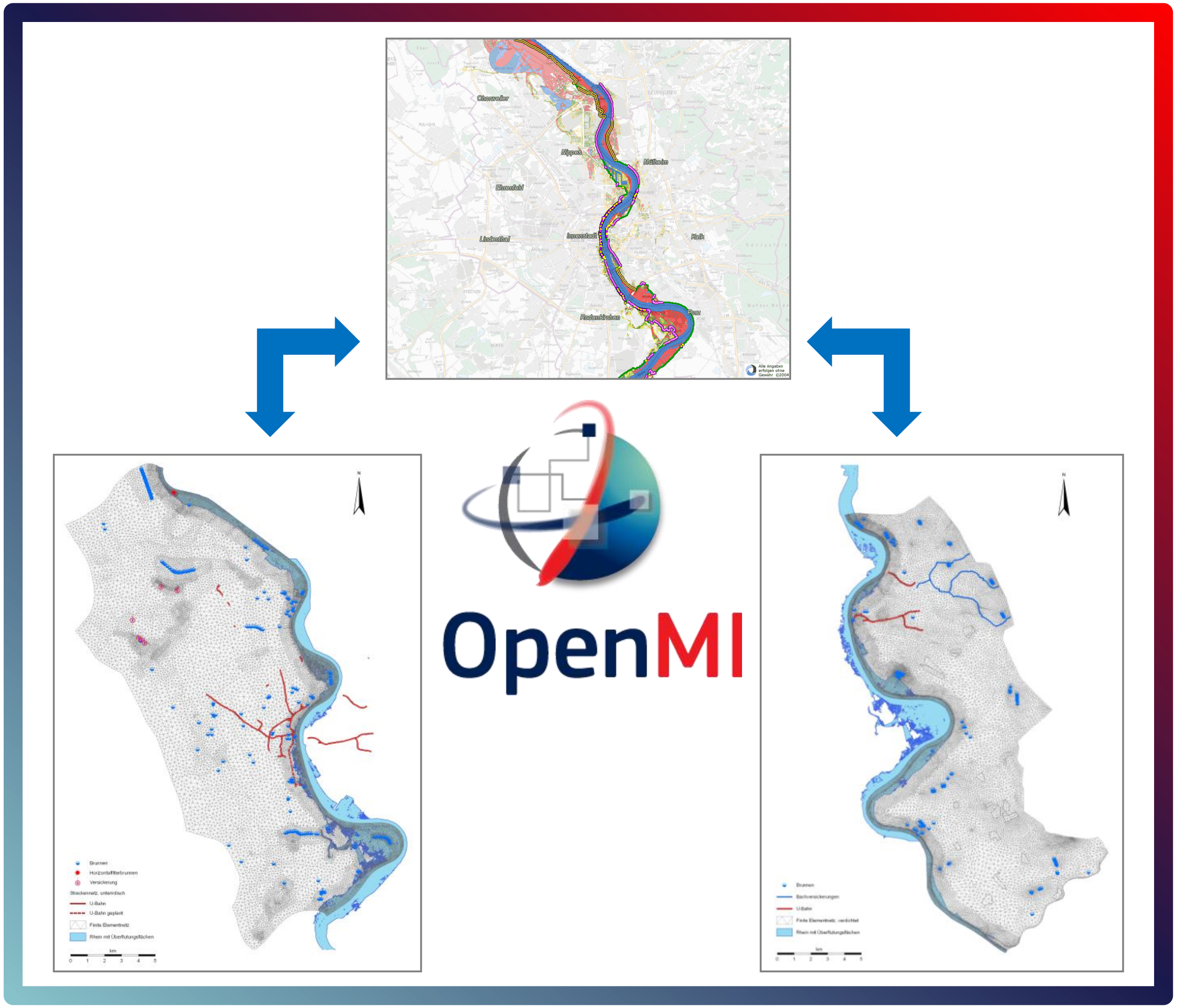
Dutch Large Scale SOBEK model



- Water level differences: less than 3 cm
- Discharge differences: less than 1 m^3/s
- Computational effort of explicit coupling increases disproportionally with the number of sub-models

→ implementation of OpenMI standard into Sobek has high potential to increase performance

Cologne: subsurface flood hazard modelling



Models

Surface water:

Sobek (Deltares)

Groundwater:

Feflow (DHI-Wasy)

Project contractor

RWTH Aachen University

Deltares

Some things to be considered …

Different exchange items for different coupling tasks

- surface water – subsurface water interaction: grid-based
- rainfall-runoff – channel flow: node based

State-of-the-art code performs better than legacy code

- RTC-Tools (direct access) vs. Sobek (client-server-technique)

Delft-FEWS is *not* model coupling in terms of OpenMI

- FEWS: sequential coupling
- OpenMI: online-coupling on time-step basis, parallel / simultaneous simulation

OpenMI 2.0 provides a loop approach (kind of parallel coupling)


Outlook

High potential for coupled flow simulation across

- country borders (e.g. Germany, The Netherlands)
- institution borders (two neighboured water authorities)
- software producer borders (Deltares-DHI, Deltares-Alterra)

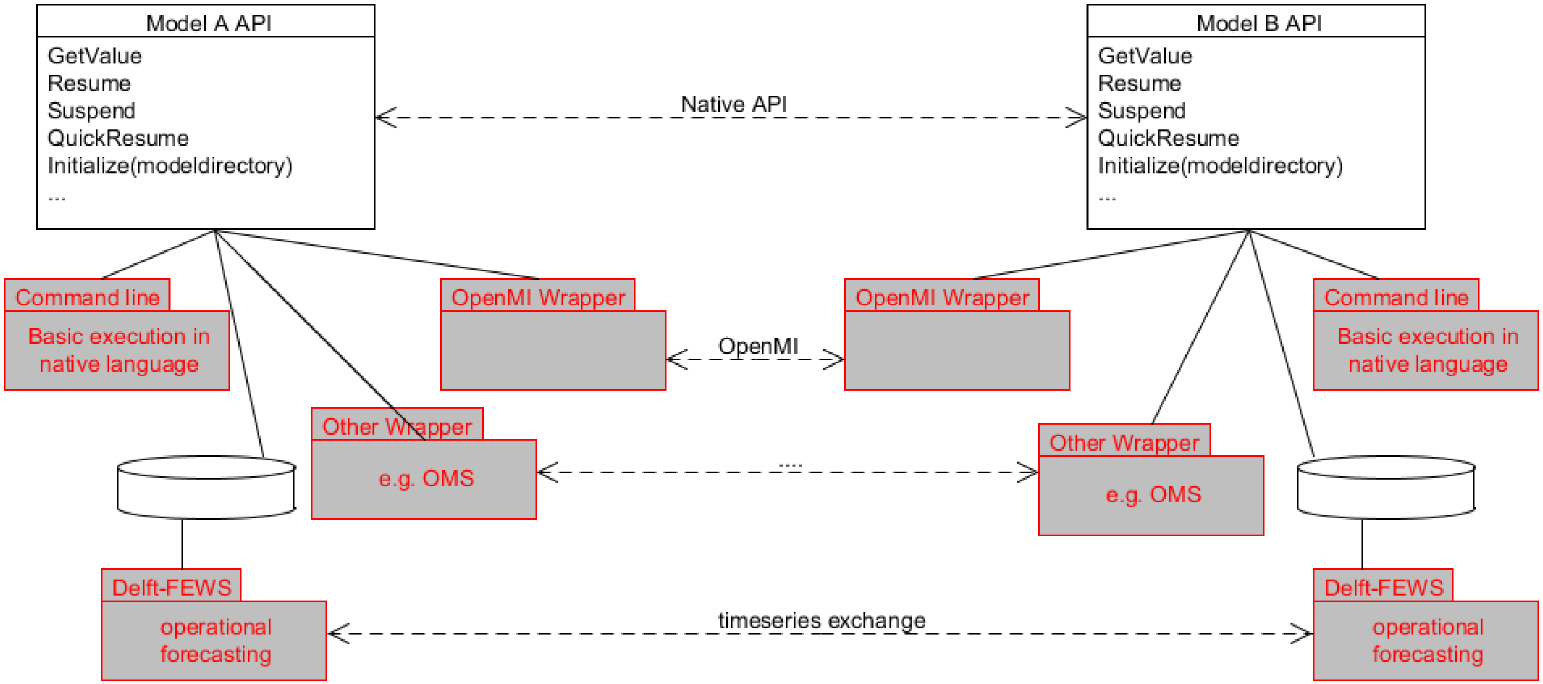
For coupled processes model coupling is already frequently applied:

- DeltaShell, Sobek 2, OpenStreams …
- OpenMI-coupling can be a first step for more: RTC-Tools is now integrated in DeltaShell

Next steps:

- Bring the OpenMI technique to consultants, universities and other disciplines
- Get further on-the-job-experience with OpenMI 2.0
- OpenMI - CIPRNet workshop during the Delft Software Days on 27th October 2014

Current projects: OpenStreams



Deltares

Further reading

OpenMI Association (www.openmi.org)

- general information about OpenMI and the OpenMI Association
- download and documentation of OpenMI 1.4 and 2.0

Deltares OpenMI public wiki (publicwiki.deltares.nl)

- documentation, how-to tutorials

Publications

- …
- …
- …

Appendix

Appendix

OpenMI-compliant components and experts

OpenMI compliant components

(selection)

- Sobek 2 (Deltares)
- Sobek 3 (Deltares)
- Modflow (USGS)
- Wanda (Deltares)
- Feflow (DHI-Wasy)
- WFLOW (Deltares)
- RTC-Tools (Deltares)
- Waqua (Deltares)

…

Deltares colleagues with OpenMI-experience

Software developers

- Stef Hummel
- Peter Gijsbers
- Edwin Spee
- Gennadii Donchyts
- Bert Jagers

Modellers

- Bernhard Becker
- Jan Talsma
- Quanduo Gao
- Neeltje Goorden
- Geert Prinsen
- Juzer Dhondia
- …

CIPRNet

Critical Infrastructure Preparedness and Resilience Research Network

www.ciprnet.eu

Conjunctive modelling with OpenMI

Andreas Burzel and Bernhard Becker (Deltares)

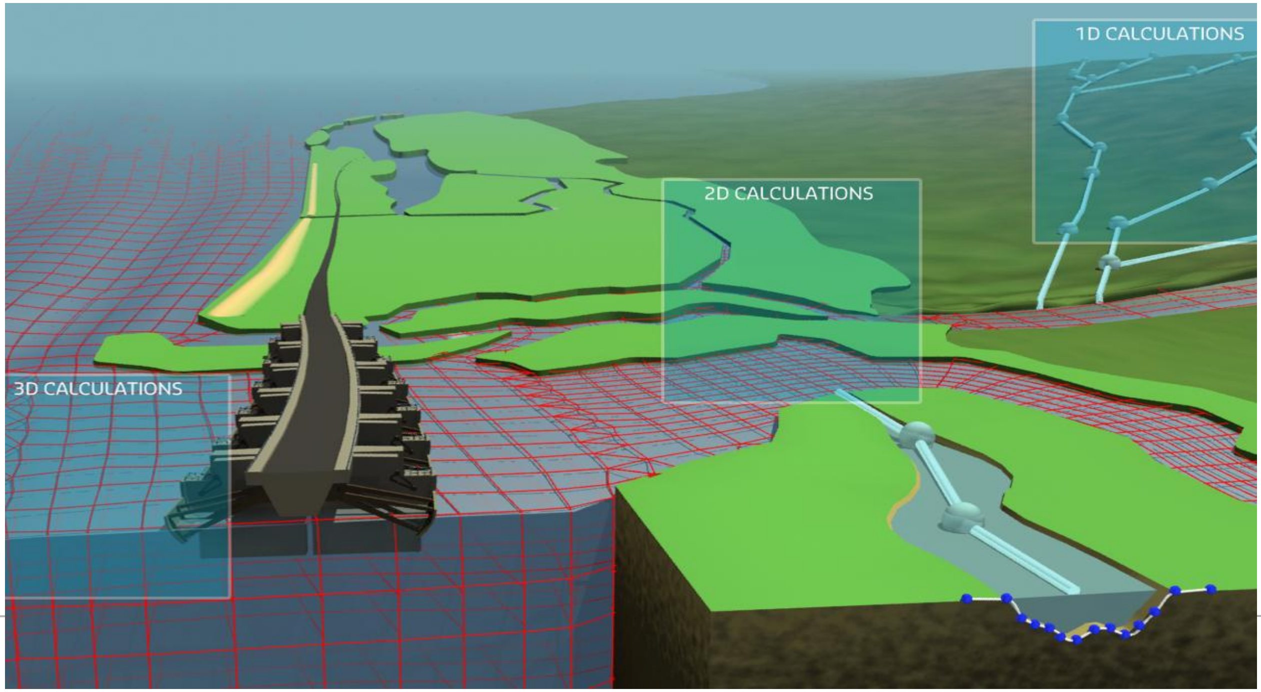
andreas.burzel@deltares.nl | bernhard.becker@deltares.nl

Modelling, Simulation and Analysis of Critical Infrastructure Training School (Edition 0)

Deltares Headquarters – Delft (The Netherlands) – 3-4 February 2014

What is a model?

Conceptual model: How does a system operate?
Mathematical model: A set of equations
deterministic (physics-based) – empirical – logical
Computer model: Coded equations
Generic model: Simulation software (GUI, input, output)
Site-specific model: Generic model + site-specific data



What is conjunctive modelling?

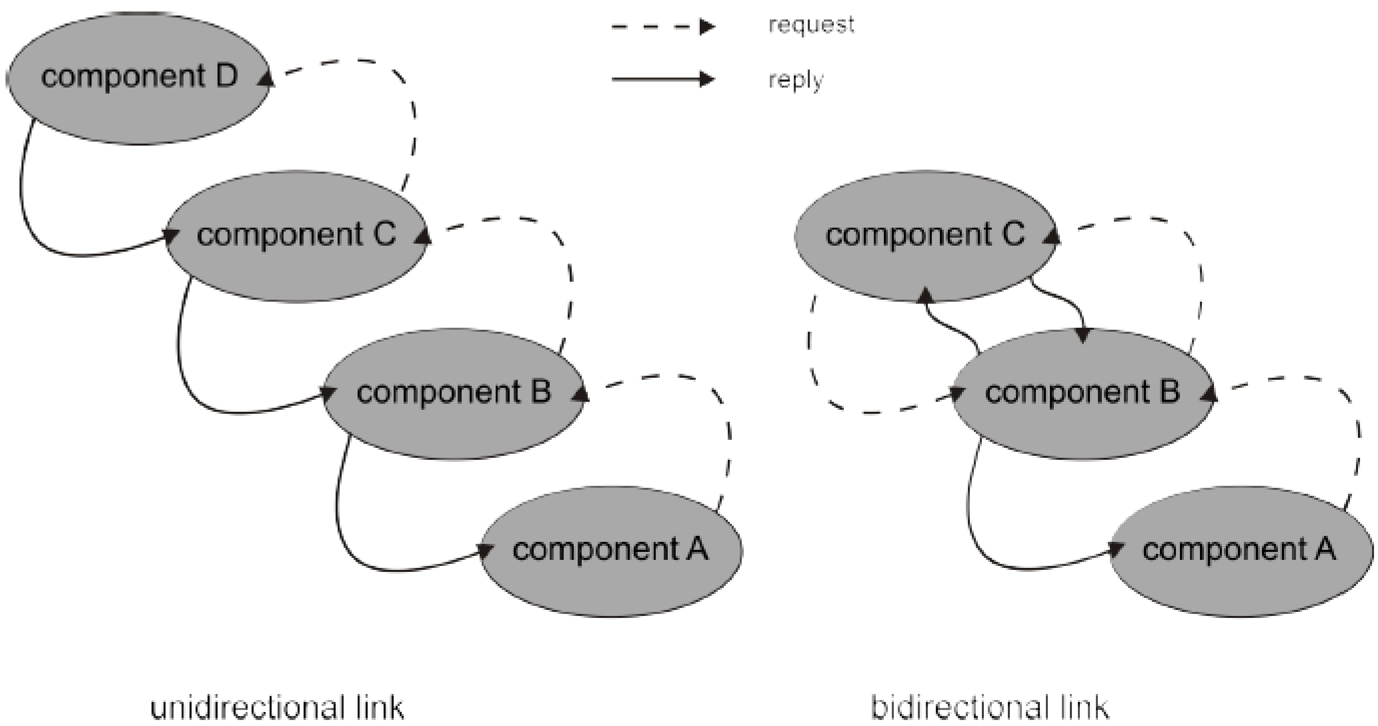
Conjunctive modeling:

- link models to model process interaction

Coupled modeling:
- data transfer in two directions.
- requires data exchange on a time step basis

Uncoupled conjunctive modeling:
- data transfer in one direction
- not necessarily o n a time step basis.

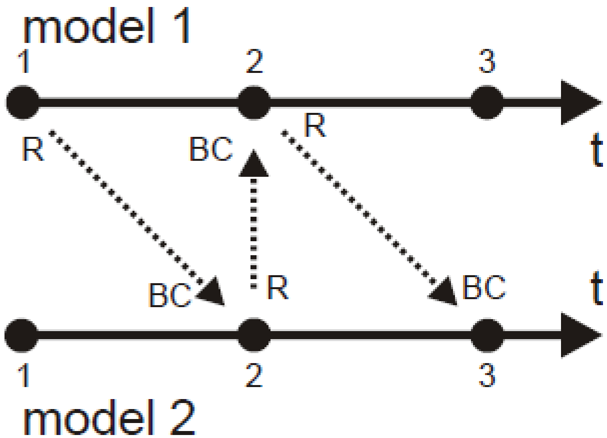
Unidirectional and bidirectional coupling

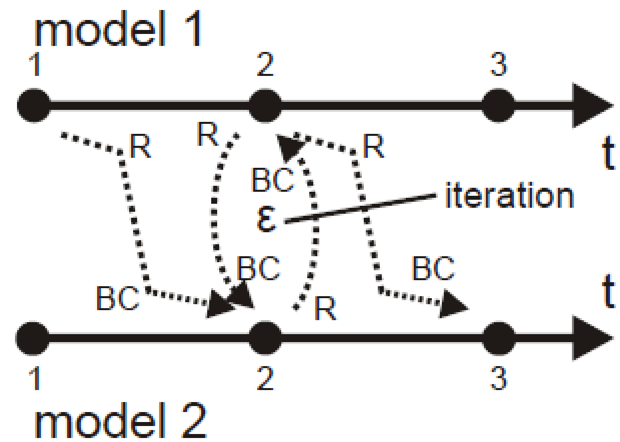


uncoupled

coupled

Model coupling





External coupling
- easy to implement
- mass balance errors

Iterative coupling
- advanced
- more accurate
- computationally more expensive

Simultaneous solution: multiple processes in one equation system
- highest level of coupling
- accurate
- time steps resolution must be the same
- equations must be of the same type

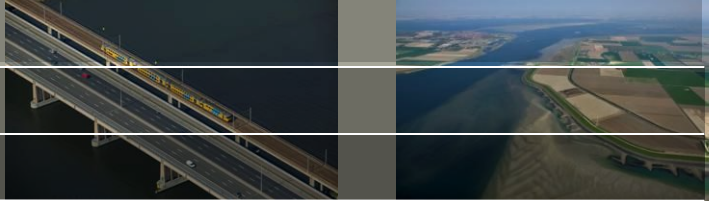

Task: conjunctive modelling

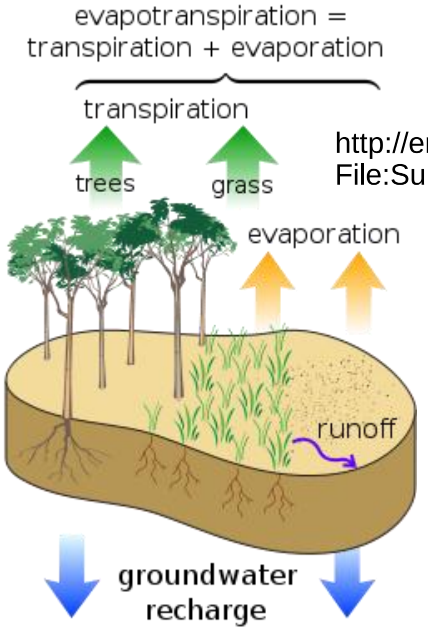
1. Compose a scenario including different possibilities of critical infrastructure failure.
2. Identify models that can represent the relevant processes
3. Draw a sketch of models and their interactions.
4. Discuss benefits of model coupling for your setup against uncoupled modeling.
5. Discuss alternative setups.

Table 1: Relevant processes and corresponding models

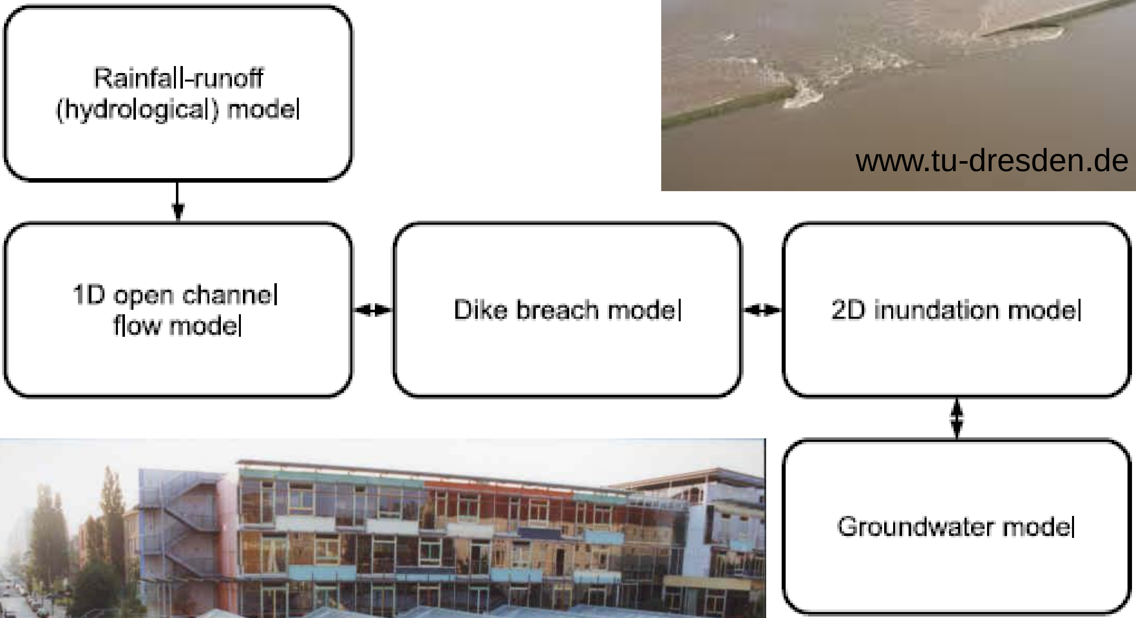
no.	process	model
1	rainfall-runoff	hydrological model
2	river flow	1D open channel flow model
3	dike breach	dike breach model
4	hinterland flooding	two-dimensional flood model
5	groundwater head rise (subsurface flood)	groundwater model

Conjunctive modeling



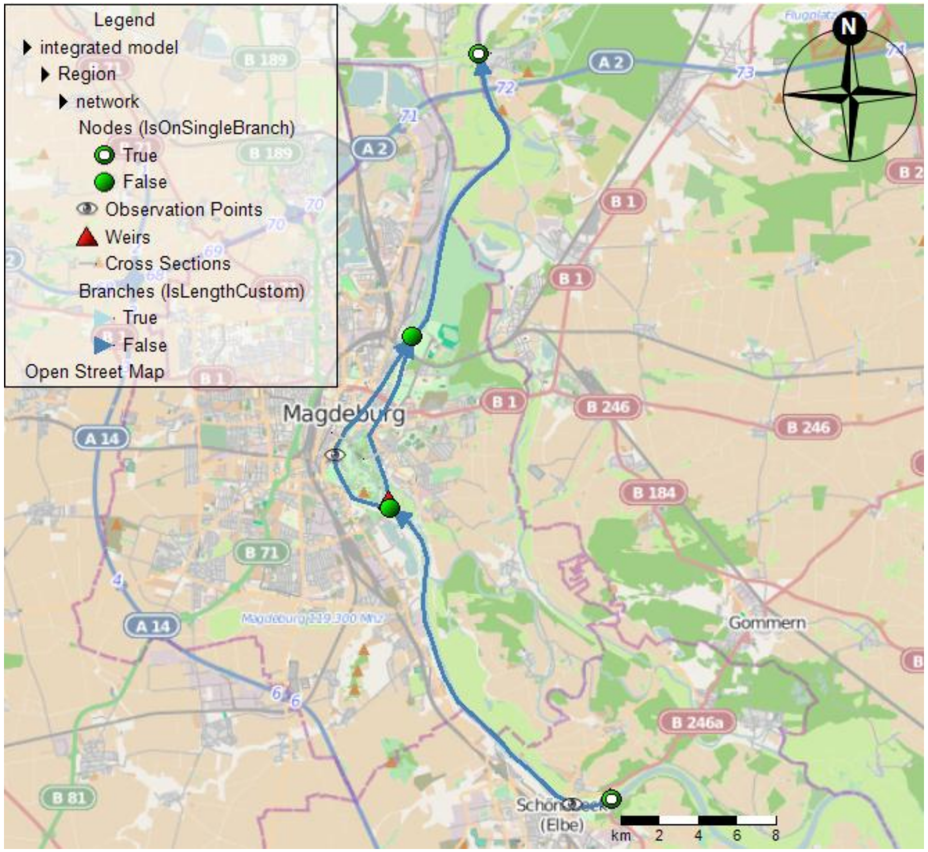


http://en.wikipedia.org/wiki/
File:Surface_water_cycle.svg



Deltares

Example: Elbe river, Magdeburg (Germany)

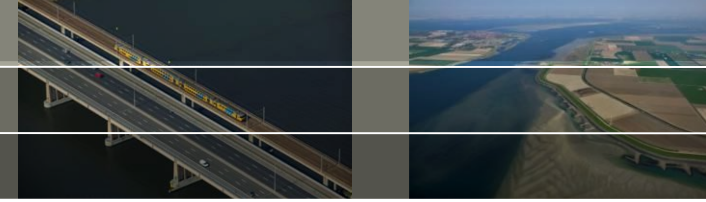


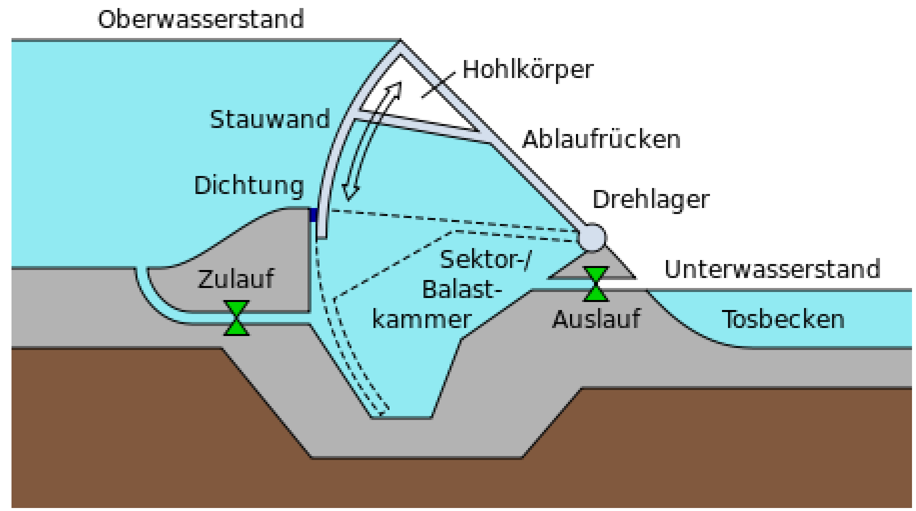
- Main station
- Railway track junctions
- Weir
- Gauges “Magdeburg” “and “Schönebeck
- Old Elbe branch
- Main river channel

Sobek model schematization
physical model (St.-Venant equations)

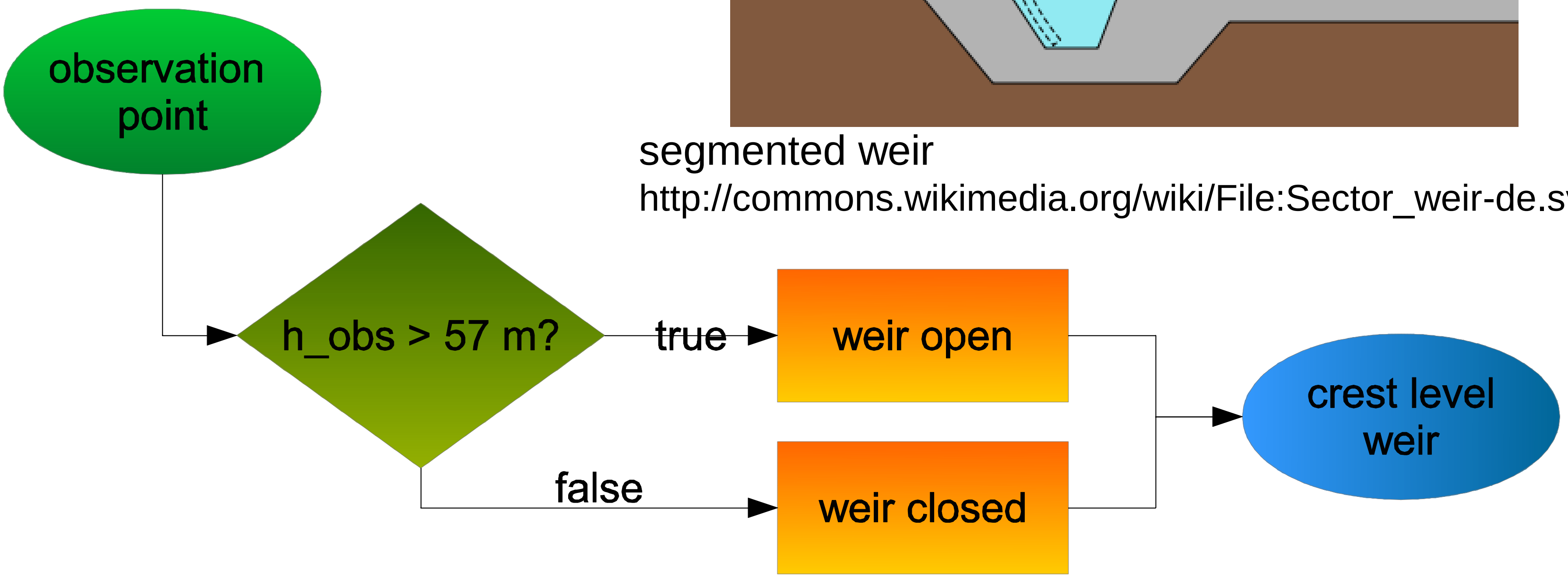
Deltares

The real-time control model





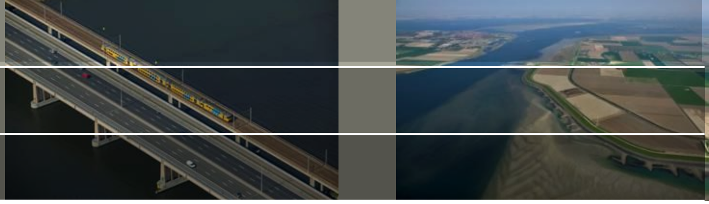
segmented weir
http://commons.wikimedia.org/wiki/File:Sector_weir-de.svg



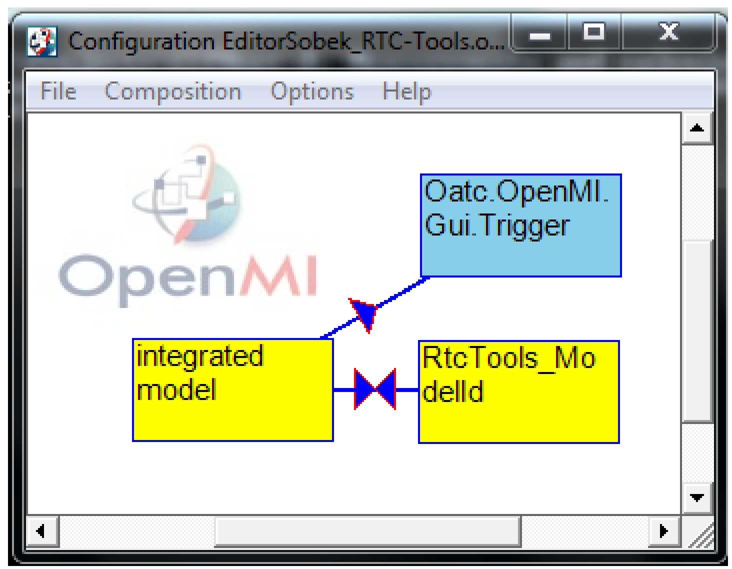
logical/relational model

Deltares

Processes and models



	Open channel flow	Human operations
Simulation programme	Sobek	RTC-Tools
Output parameters	Water level, discharge	Crest level
Input parameters	Crest level	Water level



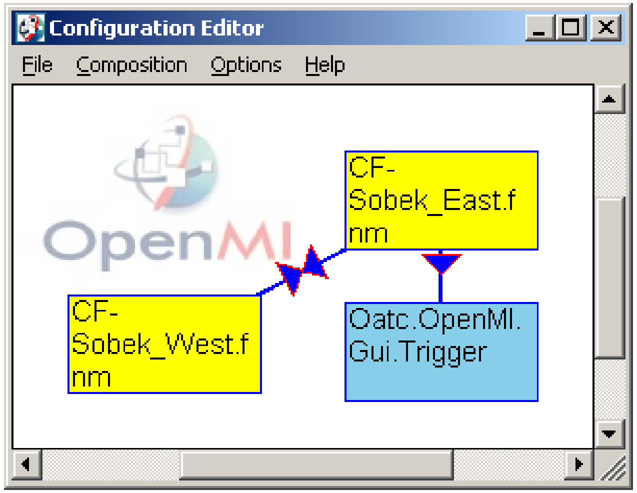
Setting up an OpenMI-Composition

omi-file: the OpenMI-Compliant Component

- Where is the DLL with the computational core and OpenMI-Interface?
- Where are the input files?
- What else? (Command line arguments)

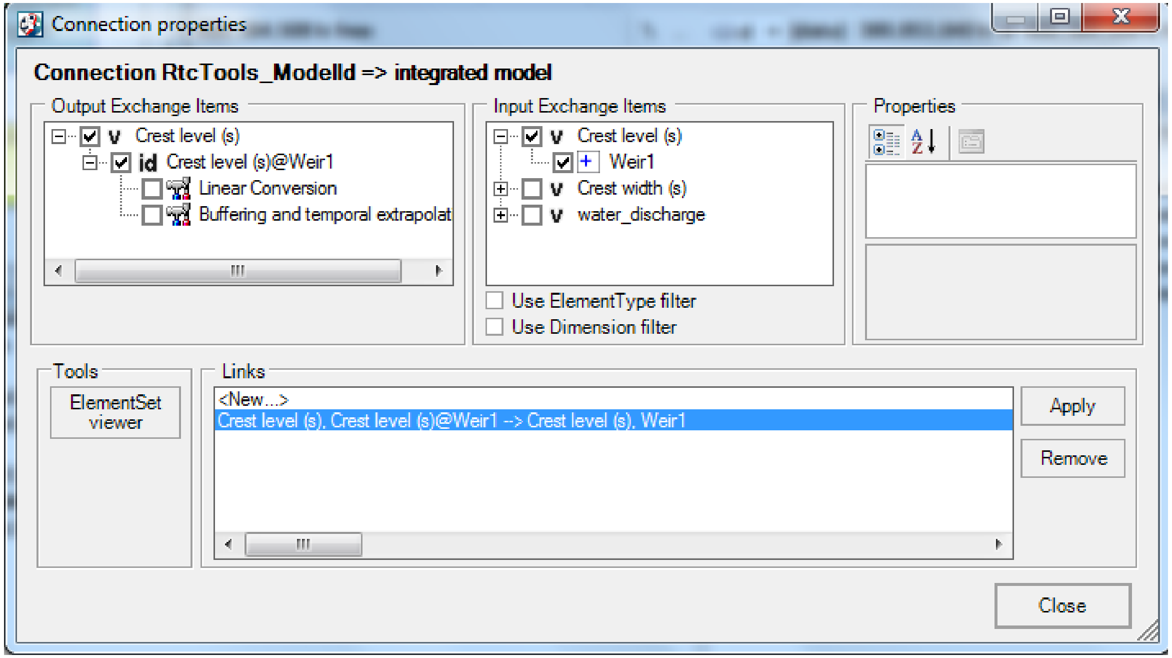
opr-file: the OpenMI-Composition

- Which components (i. e. models)?
- How coupled?
- Which simulation period?
- Where is the Trigger linked with?



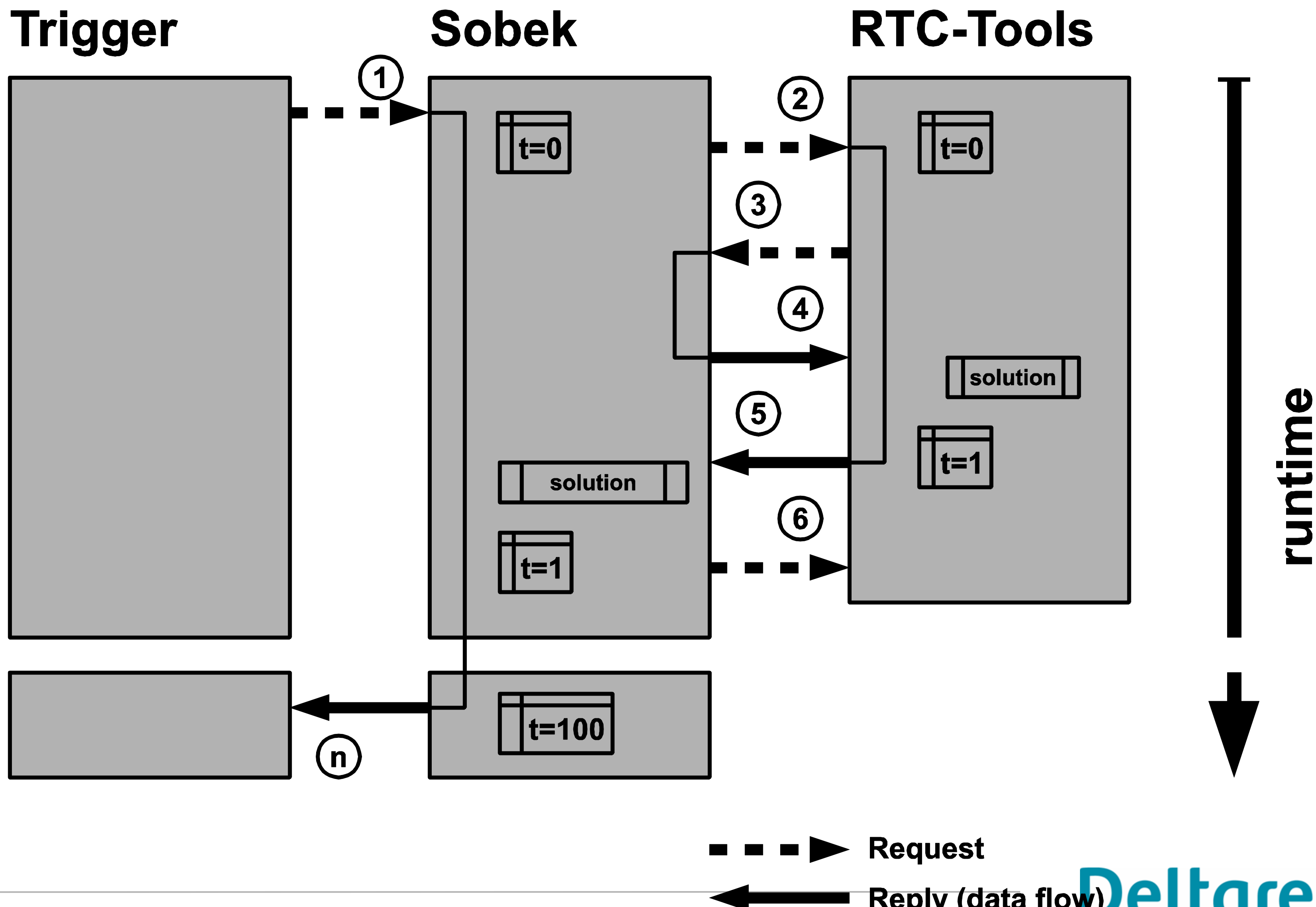
OpenMI Exchange items

What? e. g. water level in metres.
Where? e. g. Gauge Schönebeck
input exchange items: e. g. boundary conditions
output exchange items: e. g. simulation results

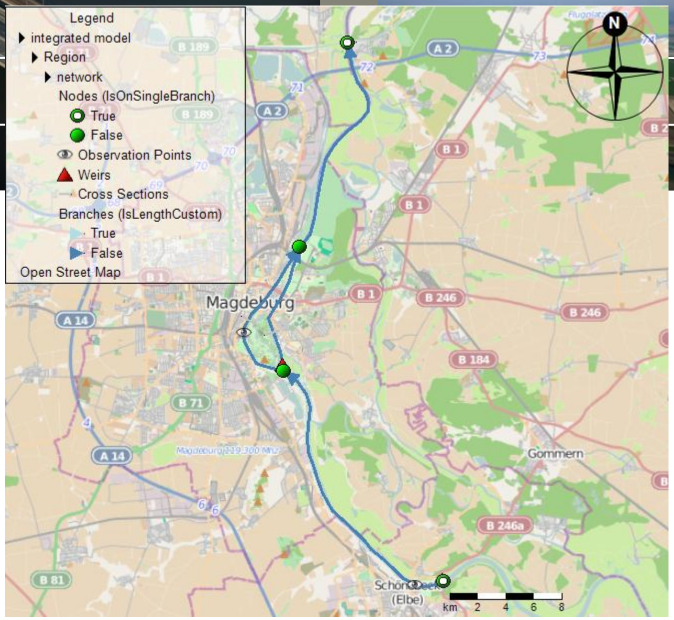


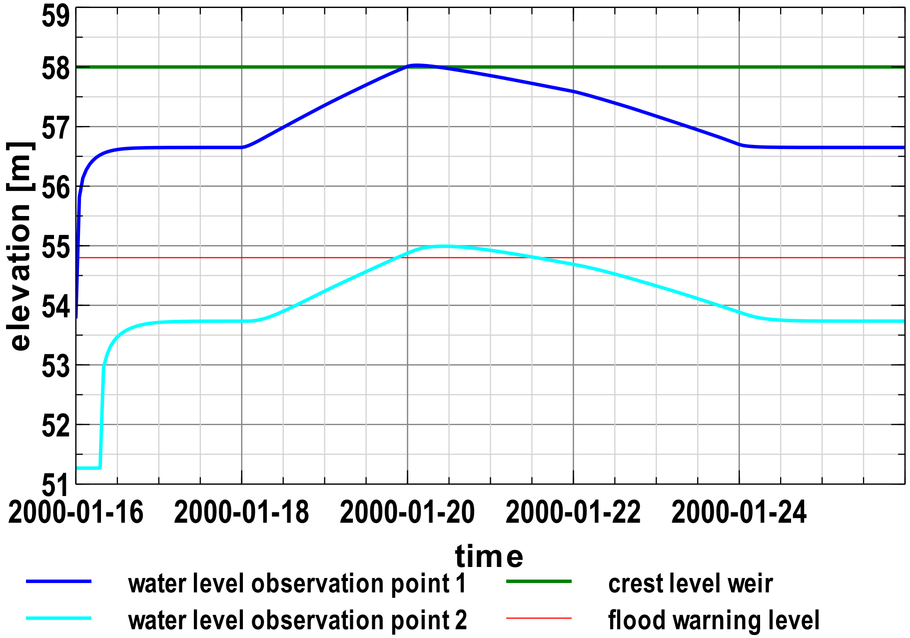
Deltares

Data exchange mechanism

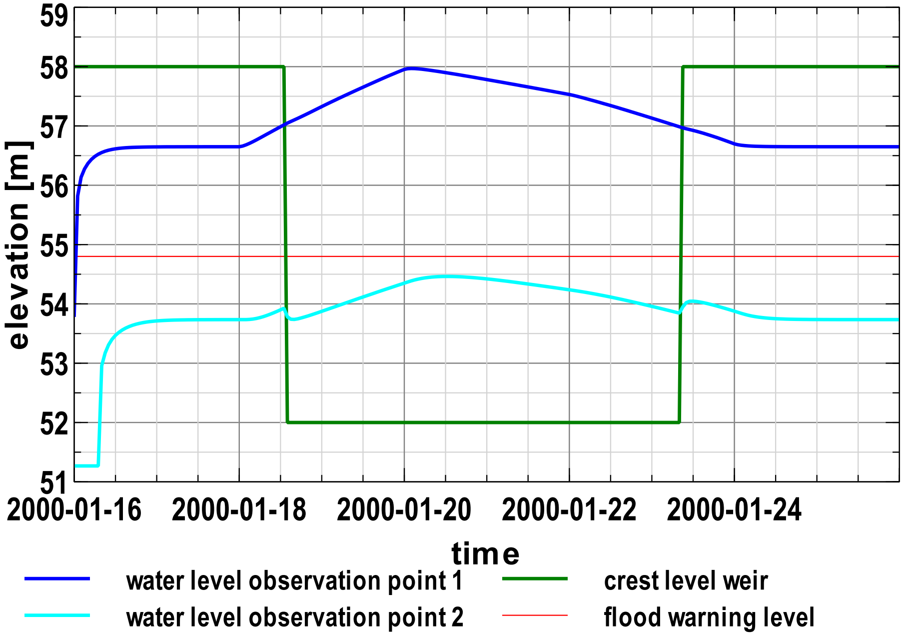


Simulation results





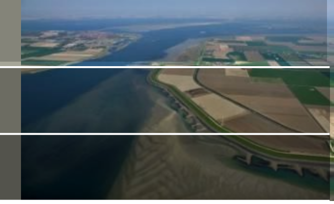
Sobek



Sobek and RTC-Tools

Deltares

OpenMI-compliance



DLL with OpenMI-functions

`Initialize()`

- read input files
- populate exchange items (e. g. water level in meters at node number 62)

`GetCurrentTime()`

- returns the current simulation time as Modified Julian Day

`GetValues()`

- returns a simulation result for an OutputExchangeItem

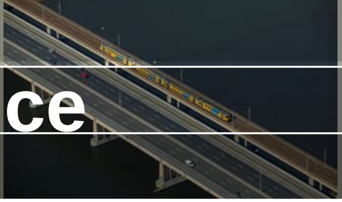
`SetValues()`

- sets a value for an InputExchangeItem (boundary condition)

`PerformTimestep()`

- solves the flow equation for **one** time step

Migration to OpenMI compliance




Re-organise the computational core

- `.exe` → `.exe` and `.dll`
- break the big loop over all time steps (t < tend)
- provide internal functions (“native layer”)
  - `ComputeOneTimeStep()`
  - `ReturnListOfNodes()`
  - `ReturnSimulationTimeInSeconds()`

Couple the computational core (engine) with the OpenMI source code (C#) via MSDN PlatformInvoke

Fill the OpenMI `ILinkableEngine` member functions

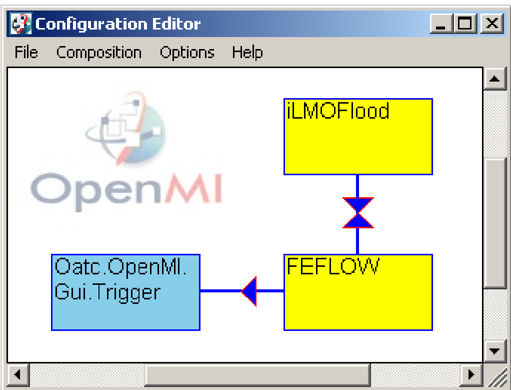
The OpenMI association provides tutorials and course material and a handbook.

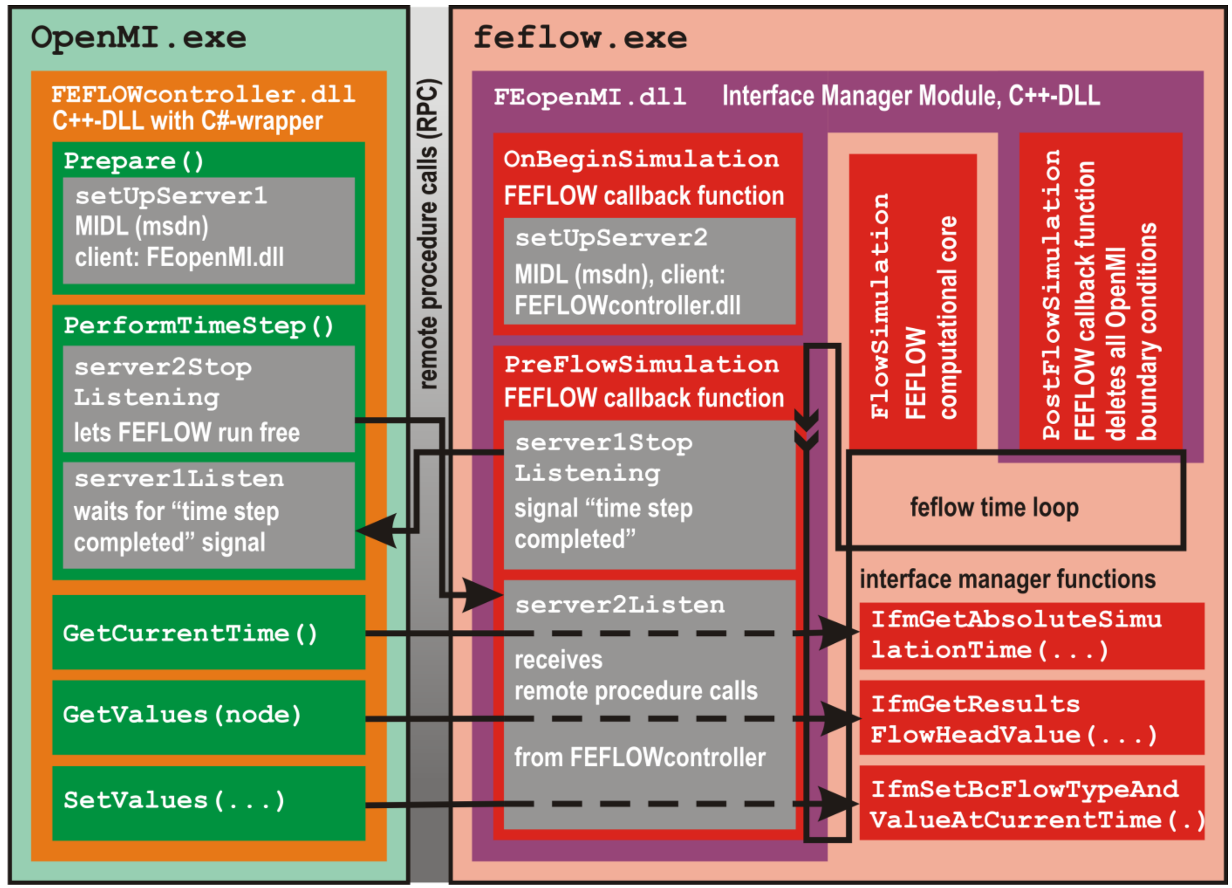
Work load:

- one week for RTC-Tools (experienced developer, “state of the art code”)
- Sobek: features are continuously developed for different cases

OpenMI compliance of FEFLOW

Feflow (DHI Wasy GmbH): control via the interface manager and remote procedure calls





Challenge:
Source code not available

Becker & Schüttrumpf, JHydroInf, 2011

Deltares