

Towards an Open Modelling Interface (OpenMI) the HarmonIT project

M. Blind¹ and J. B. Gregersen²

¹Directorate-General of Public Works and Water Management, Institute of Inland Water Management and Waste Water Treatment (RWS-RIZA), PO Box 17, 8200AA Lelystad, The Netherlands

²DHI – Water & Environment, Agern Allé 5, 2970 Hørsholm, Denmark

Received: 1 August 2004 – Revised: 1 November 2004 – Accepted: 15 November 2004 – Published: 9 August 2005

Abstract. The Water Framework Directive (WFD) poses an immense challenge to water management in Europe. Aiming at a “good ecological status” of surface waters in 2015, integrated river basin management plans need to be in place by 2009, and broadly supported by stakeholders. Information & Communication Technology (ICT) tools, such as computational models, are very helpful in designing river basin management plans (rbmp-s). However, many scientists believe that a single integrated modelling system to support the WFD cannot be developed, and integrated systems need to be quite tailored to the local situation and evolve during a collaborative planning process. As a consequence there is an urgent need to increase the flexibility of modelling systems, such that dedicated model systems can be developed from available building blocks. In the recent past a number of initiatives have been started to develop an IT framework for modelling to meet the required flexibility.

In Europe the international project HarmonIT, which is sponsored by the European Commission, is developing and implementing a standard interface for modelling components and other relevant tools: The Open Modelling Interface (OpenMI). This paper describes the HarmonIT project and objectives in general. The current progress is described. It describes the roles for different types of stakeholders in modelling, varying from software coders to non-specialized users of decision support systems. It will provide insight in the requirements imposed when using the OpenMI.

1 Introduction

In 2000 the European Parliament and Council passed the ambitious directive 2000/60/EC establishing a framework for Community action in the field of water policy, known as the Water Framework Directive (WFD). The key objective of this law is to achieve good ecological status of Europe’s water re-

sources by 2015. A key aspect of the WFD is integration. The WFD aims at integrating, amongst others: i) environmental objectives – combining quality, ecological and quantity objectives; ii) all water resources -combining fresh surface water and groundwater bodies, wetlands, coastal water resources at the river basin scale; iii) all water uses, functions and values into a common policy framework; iv) disciplines, analyses and expertise, combining hydrology, hydraulics, ecology, chemistry, soil sciences, technology engineering and economics; v) stakeholders and the civil society in decision making, etc. (CIS, 2003).

To achieve the WFD objective a number of activities need to be carried out, leading to an Integrated River Basin Management Plan (RBMP) in 2009. Such a plan will include programmes of measures, leading to the desired state of the water resources. Since the task of integration is very complex, it is believed that models and other ICT based tools are very valuable to structure and analyse all information and estimate effects of measures. It is also clear that there will not be one single instrument that can support all activities across Europe, different (spatial and temporal) scales within river basins, objectives, stakeholders, data-availabilities, (etc.). Furthermore, at different phases of the implementation different levels of detail will be required.

As a consequence there is an urgent need to increase the flexibility of modelling systems, such that dedicated model systems can be developed from available building blocks. In Europe, the international project HarmonIT, which is sponsored by the European Commission, is developing and implementing a standard interface for modelling components and other relevant tools: The Open Modelling Interface (OpenMI). In this project a number of otherwise competing software vendors have joined forces. This is considered a key advantage to achieve the objective of setting a standard.

This paper describes the HarmonIT project and objectives in general. The current progress is described. The paper focuses on the implications for different types of stakeholders in modelling, varying from system administrators to non-specialized users of decision support systems.

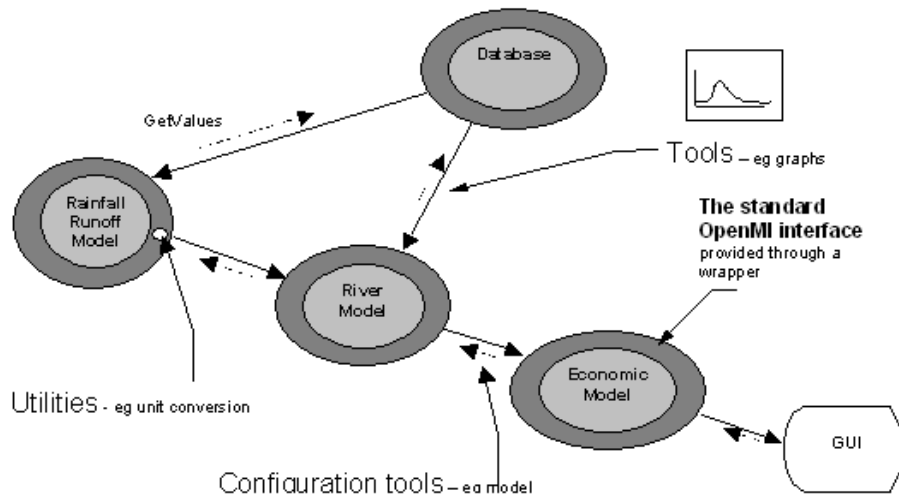


Fig. 1. Facilities for wrapping, linking and running models (Moore et al., 2004).

2 The HarmonIT objectives and benefits

The aim of the HarmonIT project is to develop an “Open Modelling Interface and Environment (OpenMI)”. The purpose of this environment is to facilitate the linking of models. This allows catchment process interactions to be represented in the formulation and selection of sustainable policies for catchment management (Moore, 2001). Such a modelling interface and environment should resolve or improve a number of complicated linkage issues, such as for example: difference in spatial and temporal scales, feedback loops, differences in spatial and temporal concepts (distributed vs. lumped, steady state vs. dynamic), different units and naming of variables, distributed computing, etc.

Much knowledge and value is already implemented in existing (integrated) modelling systems. A key requirement of the OpenMI is to facilitate, with limited effort, the use of such valuable legacy software within the OpenMI environment.

Though focussing on data exchange between models in a runtime environment, the OpenMI standard is not limiting: linkages to databases, (real-time) monitoring devices are obvious uses of OpenMI. It can also be used for linking to end-user dedicated user interfaces, and be the base of gaming tools, assessment software (etc.) as well.

For the implementation of the WFD the OpenMI standard and environment will provide the means to speed up the development of both simple Decision Support Systems for non-specialist end-users, and linking complex models to increase insight and improve results. Improving the possibilities to swap models allows changing and tailoring integrated systems to emerging needs. From a scientific perspective the OpenMI will lead to an improved ability to model process interactions, the ability to use appropriate model combinations and the ability to swap in and out different models of the same process and hence facilitate sensitivity analyses and benchmarking. The OpenMI will further lead to increased

choice for model users, an increased market for suppliers and increased opportunities for the creation of small-medium enterprises (SME's). Finally the OpenMI will allow better reuse of generic parts such as visualisation tools, thus a higher return on investment.

3 State of play and OpenMI architecture

In 2002 an overview of existing frameworks (Hutchings et al., 2002) and a HarmonIT requirements report (Gregersen et al., 2001) have been published. Based on this a framework architecture has been designed, which has been reviewed by an international panel of experts. Given some minor revisions the architecture is considered stable. The draft can be found for registered HarmonIT users on www.HarmonIT.com.

The core problem of the design was how to facilitate data exchange between models, databases and tools in a run-time environment. The solutions to this problem lie in two principles: component based design and the “GetValue” mechanism.

Component based design implicates, that all models, databases and tools must be software components. For existing models this is achieved by ‘wrapping’ the model. All components have at least the same minimum set of properties, methods and events.

The main mechanism for (run-time) data exchange is the “GetValue”-mechanism (Fig. 1). This mechanism is pull driven, meaning that a model that requires input asks a providing model for a valueset for a given time at (a set of) location(s). Three types of valuesets are defined: scalarsets (doubles), vectorsets or stringsets. The providing model calculates the required values and returns them. The GetValue mechanism is also used when getting data during e.g. initialisation of a model or retrieving data from databases or monitoring stations. Generic tools, such as for example a graph

tool utilise a “listener” mechanism: models signal when values are available and based on such signals generic tools use the GetValue mechanism to retrieve the required data.

The overall architectural design is layered, consisting of the standard itself (system layer), and three supporting layers: utilities, configuration and tools:

- The system layer consists of a description of the mechanisms, all interface definitions and definitions to describe the models and exchange variables. In principle, this is all you need to work with the standard.
- Configuration files for the integrated model system can be created based on the system layer alone. To ease this process the configuration layer has been designed, which provides the tools that facilitates linking.
- The utilities layer is independent of the architecture as well, and may or may not be used by software developers. The layer does however contain a huge amount of functionality that is especially useful when existing (legacy) software needs to be adapted to work within an OpenMI environment. Adapting legacy software to the OpenMI is called migration, and typically the existing code will be “wrapped” with some OpenMI compliant code. To keep the work on the legacy code to a minimum, the utilities layer provides a SmartWrapper, which contains helpful functionalities such as bookkeeping of links, unit conversions, buffering, time and space interpolations/extrapolations, etc. (see also Sect. 5)
- The tools layer consists of a number of useful tools, such as visualisation tools, logging tools, optimisation controllers, iteration controllers etc. Here again, the OpenMI user is free to use any (OpenMI-compliant) tools.

In 2003 detailed designs of all layers have been developed, and core principles of the design have been prototyped. The first two tests concerned the functioning of bi-directional links in hydrological modelling. The more advanced of these tests consisted of splitting a river model in an upstream and a downstream model, which is visualized in Fig. 2. At the same time step the downstream model requires inflow from the upstream model, whereas the upstream model requires water level from the downstream model for calculations. Obviously, linking the models leads to the situation that both need input from the other model at the same time. To avoid a deadlock a model can provide an initial estimate allowing computations to start, and through iterations the system should approach the required computational and hydrological stability. The test showed, that in this particular case the difference between the original model and the linked parts were negligible, even without iterations. We find this result very promising: bi-directional will become increasingly important in integrated systems: Links between surface and groundwater models, river bedding plant growth model and

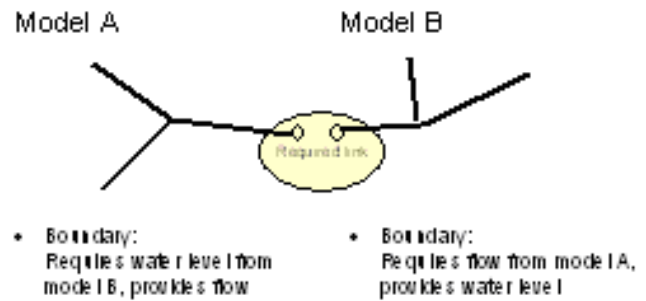


Fig. 2. Visualization of the bi-directional link in the case of a split 1 dimensional flow model.

water level river model, sediment model and river model are other examples of illustrating the use of bi-directional links.

A third test involved running a distributed linked system over four computers, consisting of three models (each from a different partner) and a generic graph tool. This test has been successful, too, though the bi-directional link described above could not fully be tested due to technical limitations of the developer environment (.Net). We consider bi-directional links over multiple machines to be of less importance for practical situations. Furthermore, the problem encountered is a technical implementation problem, and independent of the OpenMI standard.

In this year (2004) the OpenMI is implemented, including utilities such as the SmartWrapper, configuration tools and generic tools. A number of models are migrated. Much work is carried out to provide guidance on how to use OpenMI. This guidance will be tested from December 2004 onwards, when a number of project partners not involved in the current implementation phase, will migrate a number of their models and tools. We believe that within the project duration partners will migrate about 30 models to the OpenMI environment, covering (aspects of) hydrology (channel flow, groundwater, etc.), water quality, ecology, economy, sewers, real-time control, etc.

4 How HarmonIT will work in the future

The way integrated modelling systems are developed and tailored to the user’s needs in the future is visualized in Fig. 3. Starting from the bottom of the figure, models, tools and databases with an OpenMI interface will be developed by model and tool coders. This development will frequently be “migration”, which means that existing code is adapted, such that it meets the OpenMI Interface specifications. For the purpose of linking models it is only required to provide the following through the OpenMI interface: required inputs, outputs and some parameters for tuning integrated systems. What the responsibilities are of the model code developer is described in Sect. 5. Modelling specialists will populate and tune individual models, such as for example groundwater models. For existing models, the already available user interface for populating models will be used, since, (as

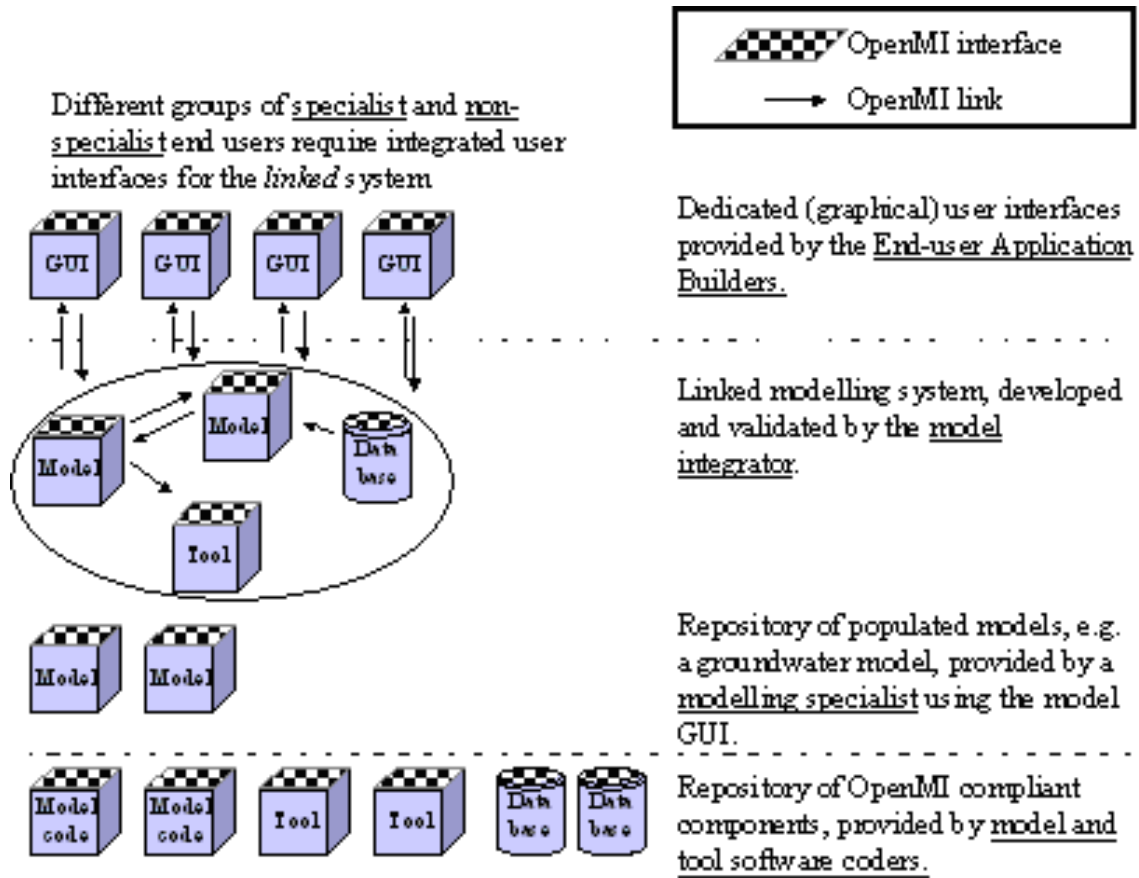


Fig. 3. Schematic of OMS implementation for natural resource modeling.

stated above) not all variables may be accessible through the OpenMI interface.

A model integrator, who specializes in establishing scientifically sound links between models, tools and databases, will link the populated models. Since an integrated system of validated models does not imply that the integrated system is valid, the model integrator will tune the system in cooperation with the specialist on the individual models.

On top of the integrated modelling system dedicated user interfaces for both specialist and non-specialist end-users will be build by the “end-user application builder”. For non-specialist end-users this means that s/he may be able to change some settings, develop scenarios etc., and is not burdened with the working and parameters of all different models and tools in the background. We believe that the work of the end-user application builder will become increasingly important.

Not represented in Fig. 2 is the role of an organisation’s computer (network) systems administrator. Especially in the case of distributed systems, the OpenMI environment needs to be able to cope with a variety of networks and their security properties.

5 The responsibilities of the model code developers

The model code developer’s foremost responsibility is to ensure the compliance with the OpenMI standard. But, complying with the technical OpenMI is not the only task of the code developer. We assume, that the model code developer is also the expert who knows best how an existing system needs to be migrated. First of all s/he needs to decide if individual functional components or the entire model as such is migrated. Conceptually, and for parsimony reasons small functional units are preferred, but economic, commercial and practical reasons may lead to larger components (which internally can again be OpenMI compliant). Second, s/he needs to decide which data need to be exposed to the outside world both from the viewpoint of providing and accepting values. This work will usually also involve the modelling specialist. Which data need to be exposed is very dependent on the use of the model component. In many cases it will be sufficient to expose data that need to be exchanged by models. But if scenarios need to be calculated it may be necessary that underlying schematisations can be altered using the OpenMI interface. For example, widening a river’s bedding and changing its overgrowth is a strategy to reduce flood risk. An end-user decision support system should thus allow calculating the effects of such measures. This means, that the

schematisations of individual models need to be adaptable, e.g. the cross-sections and riverbed roughness parameters. There are different strategies to accomplish this and we believe that more and more features of models will become accessible through the OpenMI interface. It remains the responsibility of the code developers to decide if and how such data can be altered in an OpenMI environment.

Though the OpenMI design allows all models to calculate at their own time stepping and spatial representation, the Get-Value mechanism requires that output can be delivered at any given time and place (reasonably within the models overall temporal and spatial boundaries). The model coder needs to make sure that interpolation and extrapolation mechanisms are in place: The HarmonIT project will provide a library of functions, which can be extended: it is the responsibility of the model coder to decide which functions interpolation and extrapolation functions are appropriate.

To achieve this the model needs to know at what time step it is. Based on this knowledge the model component may

- Start calculations, if the required time step has not been reached but can be computed.
- Extrapolate in time if the required time step cannot be reached.
- Search its buffer (if implemented) if the requested time step has passed, otherwise calculate required values)
- Interpolate between different time steps if the requested time step does not match the internal time stepping.
- (Dis-) aggregate in space and time, if the requested spatial and temporal representation does not match the internal representations.

To support the model code developer the SmartWrapper utility will provide a library containing some basic functionality to support the actions listed above. It is up to the model code developer to use the library or implement desired functionality.

In advanced integrated systems iterations and optimisations may be required. To facilitate efficient computing it is recommended that models are able to revert to an earlier state in the calculations. Again, it is the model code developer's responsibility to implement such functionality.

6 Discussion

The HarmonIT team feels that it is progressing quickly in its effort to develop a standard for exchanging data in integrated water management. The HarmonIT initiative is all but alone in its effort. Dozens of initiatives are ongoing to build modelling frameworks and integrated systems. See for reviews (Moore et al., 2004; Whelan and Nicholson, 2000). None of the existing frameworks appears to be successful in achieving the status of a de facto standard. Why do we believe that the HarmonIT initiative is not just another attempt?

1. First of all, three commercial, otherwise competing organisations in hydrological modelling have joined forces and invested quite large amounts of their own R&D funds. Together these organisations have a significant share in the market of hydrological and environmental modelling. The three organisations and many of the other project participants have significant expertise in integrated modelling, integrated modelling systems and IT frameworks. It is in their interest to use the standard, given the investments.
2. Secondly, the value of existing software at the participating organisations is extremely high, and all share the problem of increasing (customer) needs for flexibility, rising maintenance costs and the need to collaborate. As a result, the effects of the OpenMI design on existing software and on future requirements and development opportunities are continuously points of attention. A tangible result of this attention is the development of the SmartWrapper, which will support migration towards the OpenMI standard, making it both time and cost efficient.
3. The HarmonIT team spends much time and effort in documenting the developments from the perspective of different interest groups, making the OpenMI as transparent as possible and providing tailored guidance.
4. The HarmonIT team is actively preparing for an organisational structure to support OpenMI beyond the project's time line.
5. The designs are open to input from the outside world at early stages.
6. A significant number of well-known models will comply with the OpenMI at the project's end.
7. If required, the OpenMI system may be part of a larger system, and vice versa, existing (larger) frameworks and integrated modelling components may be included in an OpenMI environment as a single component.
8. Last, but certainly not least, the project is supported and reviewed by a number of leading experts in the field from all over the world.

We feel that the OpenMI architecture and associated tools and utilities will be an attractive way for model providers and model users to create systems for integrated catchment management.

We believe that on success OpenMI will be opening ways for advanced environmental research. The research community may benefit strongly: the water management community may adopt new scientific achievements much faster. Many aspects of integrated systems, such as the effects of integrating different spatial and temporal scales or modelling concepts can be researched in-depth. Helpful tools will make a researchers work more and more productive.

Acknowledgements. The authors would like to thank their fellow partners in the HarmonIT project, originating from 14 different organisations, for their help and support in preparing this paper.

For more information please access the project's pages on www.HarmonIT.com.

Edited by: P. Krause, S. Kralisch, and W. Flügel

Reviewed by: anonymous referees

References

- CIS (Common Strategy on the Implementation of the Water Framework Directive): Best practices in river basin planning Work Package 2: Guidance on the planning process, 2003.
- Moore, R. V.: Description of work for the HarmonIT project agreed during contract negotiation, IT Frameworks (HarmonIT), Proposal number: EVK1_2001-00037, www.HarmonIT.com, 2001.
- Moore, R., Tindall, I., and Fortune, D.: Update on the HarmonIT project – The OpenMI Standard for model linking, Proc. Hydroinformatics 2004, Singapore, Vol. 2, pp. 1811–1818, 2004.
- Hutchings, Ch., Struve, J., Westen, S., Millard, K., and Fortune, D.: State of the Art Review, Work Package 1, IT Frameworks (HarmonIT) Contract EVK1-CT-2001-00090, http://www.harmonit.org/docs/repu_01_09_soa_review_approved.pdf, 2002.
- Gregersen, J., Malmgren Hansen, A., Michael, L., and Enggrob, H. G.: Requirements report, Work Package 2, IT Frameworks (HarmonIT) Contract EVK1-CT-2001-00090, http://www.harmonit.org/docs/harmonit_wp2_final_report.v2_approved.pdf, 2001.
- Whelan, G. and Nicholson, Th.: Proceedings of the Environmental Software Systems Compatibility and Linkage Workshop, NRC Report Number: NUREG/CP- 0177 <http://www.nrc.gov/reading-rm/doc-collections/nuregs/conference/cp0177/0177.pdf>, 2000.