

Table 1. Progression of numerical model code structure

Single code	Multiple sub-routines	Self-contained and reusable objects	Component-based models
Line 1	Main	Object 1	Model 1
Line 2	Subroutine 1	Object 2	Model 2
Line 3	Subroutine 2	Object 3	Model 3
...

The increasing need to model systems, rather than just single processes, has led to different methods for combining models of different processes. Lu & Piasecki (2012) identified the following four categories:

- (1) The use of a range of models of a geographical area. This approach does not attempt to link models, but compares and contrasts the results from different models and approaches to help improve the models themselves and overall understanding.
- (2) The construction of a monolithic 'super model' containing multiple processes as multiple sub-routines or sub-models that can be included or excluded as necessary. These models can be constructed by a number of research groups and proposed changes are typically submitted through a version control system for validation and approval before release. Examples include MIKE by DHI (<http://mike.bydhi.com>) and the Weather Research and Forecasting model (Janjic *et al.* 2010).
- (3) The use of a generic component-based modelling framework, such as the Community Surface Dynamics Modelling System (<http://csdms.colorado.edu>), Programme for Integrated Earth System Modelling, PRISM (Valke *et al.* 2006) and the Earth System Modelling Framework (www.earthsystemmodelling.org).
- (4) The use of a coupling framework to develop a community modelling system, by providing software to assist with linking different models during run-time. Coupling frameworks ensure that data can be transferred between different component models at known times and places. A good example of this approach is the Open Modelling Interface (OpenMI) standard (as described later in this paper).

All four methods are in use and there remains the issue of whether one should simply build larger monolithic models (one form of 'integration'), or use existing models and link them together – as implied by the concept of model fusion. One could of course argue that linking at the sub-routine

level is a logical extension of the concept of integration and, for this reason, many practitioners are still not convinced of the benefits of the dynamic linking of models.

However, most of the established hydraulic and morphodynamic models were developed as proprietary or bespoke packages. These models largely followed the closed architecture philosophy, which emerged in response to the development of a niche market for modelling services but which also posed problems for the clients of these services (Khatibi *et al.* 2004):

- Organizations that use or manage the results of these models need to manage the risk that a software capability becomes obsolete.
- Consultants need to maintain a store of proprietary software products.
- Each of these products contains similar tools that produce similar results but cannot be transferred from one software product to another.
- User-designed systems are not possible, as clients cannot select which tools they want to use.

Khatibi (2003a) proposed that 'an understanding of the interfaces between the interacting systems is the key for systemic problem-solving', while Khatibi (2003b) promotes a move towards an open architecture which uses standard interfaces to link modules. Khatibi *et al.* (2004) promote the use of both an open architecture and open source software. The benefits of open source software have also been discussed by Harvey & Han (2002) and others.

This paper explores many of the current issues that need to be overcome to promote the concept of model integration by the dynamic linking of models. It starts by introducing two frameworks that have been used to assess the progression in numerical modelling (largely in hydraulics) and considers their application to the modelling of integrated environmental systems. It then discusses the increasingly blurred line between observations (data) and models, before describing the evolution of OpenMI and the OpenMI standard (Gregersen *et al.* 2007). The FluidEarth initiative, which promotes a user community and provides software tools for implementing the OpenMI standard, is

then described, before the paper discusses the challenges that must be addressed to move forward from today's integrated models towards model fusion. The *Concise Oxford English Dictionary* defines fusion as the 'fusing, melting, blending of different things into one', so it might be argued that it should involve more than just the ability to link models and this paper looks at what other factors may be involved in this concept.

Frameworks for assessing model development

The development of numerical modelling systems has never occurred in isolation as a purely technical development of code. The development of numerical models is inextricably linked to the development of different types of code, changes in hardware, peripherals and operating systems. Moreover, it is also depends on the increasing number of stakeholders whose decisions may be influenced by the results from modelling and their interactions with the innovators, developers and users of numerical models.

This paper considers the historical development of numerical modelling as described by Abbott (Abbott 1991) and Khatibi and colleagues (Khatibi 2001, 2003a, b; Khatibi *et al.* 2004).

Abbott's five generations

Abbott (1991) considered there to be five generations of hydraulic models:

- (1) Numerical solutions to algebraic equations (1950s)
- (2) Project-customized modelling and the development of modelling groups (1960s)
- (3) Introduction of modelling systems. Modelling is undertaken as a service by specialized centres (1970s and 1980s).
- (4) Development of software tools as packaged products, with help, support, pre- and post-processing tools (1980s to 2000s)
- (5) Development of hydroinformatics tools (2010s) aimed at 'raising the level of discourse of its clients' (Abbott 1991), which developed into 'making electronically encapsulated modelling knowledge available over the internet' (Abbott *et al.* 2006) or 'Software-as-a-service' (Abbott & Vojinovic 2009).

The dates in brackets are approximations of when each generation rose to prominence. The level of computational hydraulic knowledge required of the typical user has fallen dramatically between generations (Abbott 1991; Abbott & Vojinovic 2009). This carries risks, which are mitigated by reductions in the freedom given to users by providing compiled code with error checks, help and support. As you

move from generation to generation, the roles of scientist, developer, user and supporter have gradually split and become distinct. The role of the common user in each generation is illustrated in Table 2 (Abbott 1991; Abbott & Vojinovic 2009; Khatibi *et al.* 2004).

The re-definition of the fifth generation as 'Software-as-a-service' (Abbott & Vojinovic 2009) represents a logical continuation of the changing role of the user, who no longer needs to be an expert to run a piece of software. As such, it is up to the developer to produce a service that has sufficient help and is sufficiently constrained to be used by a competent professional.

Systemic knowledge management

Khatibi *et al.* (2004) point out that the causes of the changes from one generation of models to another are not readily identifiable in Abbott's account and preferred to apply 'systemic knowledge management' (Khatibi 2003a, b) to explain the current status and potential future for software tools in hydraulic modelling. Khatibi (2003a) developed systemic knowledge management as a methodological tool for assessing the development of a scientific paradigm, in this case hydraulic modelling. Systemic knowledge management was derived from: (i) the concept of paradigm and paradigm shift (Kuhn 1962; Khatibi 2001); (ii) systemic problem solving developed in systems science (von Bertalanffy 1951); and (iii) knowledge management from management science (Nonaka 1998).

In systemic knowledge management, any theory or concept can be viewed as a paradigm that shifts through 'pre-paradigm', 'forming', 'proliferation', 'norming' and 'performing' stages (Khatibi 2003a). The different stages were derived from the model developed by Tuckman (1965) of how group behaviour evolves (though Tuckman's 'storming', as in 'brainstorming', has been replaced by the rather more prosaic 'proliferating'). It is interesting that a model of social interaction is used to define the stages a scientific paradigm evolves through. This emphasizes the social nature of science, including software evolution, where scientific paradigms are often conditioned by the social outlook of their time (Prigogine & Stengers 1985). Over development timescales, this is reflected in the many communities of practice that form around different software and between modellers of the same phenomena.

Each stage has different characteristics:

- **Pre-paradigm.** A range of disparate and rudimentary approaches may be developed to begin to tackle an issue. Efforts are often in isolation, may appear random and offer no competitive advantage.

Table 2. *Role of the user in generations of modelling software*

Generation	Date	Description	Role of user
1st	1950s	Numerical solutions to algebraic equations	User is dedicated professional who is also an innovator and developer.
2nd	1960s	Project-customized modelling and development of modelling groups	User is member of dedicated modelling group who innovate and develop model.
3rd	1970s and 1980s	Introduction of modelling systems. Modelling as a service by specialized centres	User need not be an innovator or developer, but is likely to work with experts.
4th	1980s, 2000s	Commercial software tools as packaged products, including help and support	User trained by model owner/agent and pays for support. User supported by pre- and post-processing tools. Reduction in required knowledge by user.
5th	2010s	Development of hydroinformatics systems (1991), software as a service (2006, 2009)	Limited knowledge of software by user – but freedom constrained by developer.

- **Forming stage**, with the development of a paradigm from the diversity of approaches around. A form of natural selection governs how models develop and determines which models flourish and which die out. As this is a social as well as scientific process, this does not necessarily mean that it is the best scientific models which win out during this stage.
- **Proliferating stage**, with either different variations of the components of the paradigm being developed, or the paradigm spreading to different disciplines. Many people adopt the paradigm. The level of organization increases, while natural selection still governs the fate of individual models. The prevalence of many viable, normally inflexible, options leads to the law of diminishing returns, with many incremental returns being repeated across systems.
- **Norming stage**, where the gains to be made from the hierarchical organization of the competing options are realized. There is a ‘conscious process of consolidation’, with an active search for synergy between components of the paradigm (that is, within a particular field – ‘longitudinal holism’) or between the different disciplines sharing a paradigm (‘lateral holism’). Partnerships between scientists, practitioners and stakeholders emerge. The principles for the performing stage are developed.
- **Performing stage**, where high performance can be achieved within a flexible environment, where custom solutions can be delivered to meet clients’ needs and the clients are aware of the limits of the system. The connections between components and the influences they have on each other are understood and utilized. There is

an interplay between technical, economic and social needs. Utility and usability are high.

Application to hydraulic modelling. Khatibi (2003b) applied systemic knowledge management to open channel flow modelling, irrigation systems, municipal water supplies and flood drainage systems, while Khatibi *et al.* (2004) applied it to hydraulic modelling software to obtain the following five stages:

- **Pre-paradigm.** Before the invention of computers a number of mathematical and empirical approaches became established in hydraulic engineering. But while this resulted in many elegant analytical solutions or data driven empirical representation, such solutions were hindered by the need for manual computation.
- **Forming stage**, with the development of inflexible project-specific codes for fundamental problems in niche markets. The speed of a numerical calculation gave it a competitive (or selective) advantage over a hand calculation. The user was a dedicated professional, who was also an innovator and developer. A form of natural selection governs how models develop and determines which models continue to be developed and which die out. (This stage has similarities with Abbott’s first and second generations.)
- **Proliferating stage**, with the development of many general-purpose, modular (often sub-routine based) software products, with pre-processing and post-processing tools. Closed architectures were used, which made it impossible to plug-in innovative components or those from other providers. (This is similar to Abbott’s third and fourth generations.)

FROM INTEGRATION TO FUSION

- **Norming stage**, where open architectures are expected to prevail, allowing interoperability between software systems and thus enabling the user to design bespoke models from a choice of components. The development of published interfaces will play an important role in allowing different models to be used within a modelling system. The current situation represents the onset of this phase, which is still punctuated by many attributes of those previous.
- **Performing stage**, where the open source movement is expected to play a pivotal role in sharing freely and improving software source code, and web services allow increased flexibility, availability and uptake. The last two stages, which concentrate on the development of integrated modelling and data services, complement Abbott’s fifth generation (Software-as-a-service).

Khatibi *et al.* (2004) postulated that software architecture, defined as ‘the conceptual structure and logical organization of a computer or computer-based system’, is the cause of the paradigm shifts observed. This is affected by developments in processors, data storage, peripherals and the user interface, as well as programming environments, standards and the rise of the World Wide Web, as illustrated in Table 3.

Software developers were able to differentiate their products in the market through their user-friendly interface or attractive post-processing graphics (as occurred in Abbott’s fourth generation or Khatibi’s proliferating stage) only after the software for these had been developed. As developments progressed, web interfaces to software were built on the standards developed for the Web.

Khatibi *et al.* (2004) promoted the use of open architecture and open source code as important concepts for moving hydraulic modelling through the norming stage to the performing stage. Figure 1 (after Khatibi *et al.* 2004) shows how the openness of an environmental modelling system’s architecture

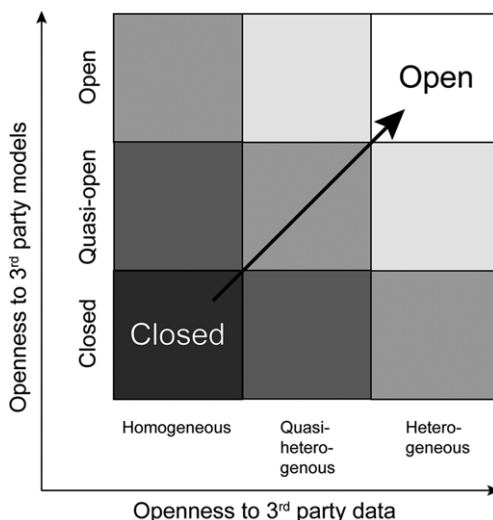


Fig. 1. Assessment of the openness of a model’s architecture.

can be assessed in terms of its openness to models and data. Its openness to ‘third-party models’ (external model components) is categorized as follows.

- An open system is one that can link to models from other software providers.
- A quasi-open system can link to a particular sub-set of models.
- A closed system can only link models from within its own system.

Its openness to other data is judged as follows.

- A heterogeneous (open) system can use datasets from third-party products.
- A quasi-heterogeneous system can use only a proprietary sub-set of data products.
- A homogeneous (closed) system accepts only native data.

Table 3. *Developments in computing environments*

Decade	Hardware	UI	Data storage	Software architecture
1970s	Mainframe	Terminals, keyboard	Floppy disk	Closed system and closed code
1980s	PC	Monitor, keyboard and mouse	3.5" floppy disk	Free software foundation, GNU
1990s	Server	GUI, WWW, Graphical output	CD	Linux, GPL
2000s	HPC	Web-style UI	DVD, USB drive	Standards (including OpenMI) open source Initiative
2010s	Cloud, grid, tablet	Semantic web, SaaS	Cloud/grid	Integrated modelling, web services

Khatibi *et al.* (2004) saw that one of the requirements for the development of open architectures would be the development and application of standards, as this would enable people to work together by establishing common rules and protocols. In recent years there has been a great increase in commonly used standards, approved by bodies such as the International Standards Organization (www.iso.org), the Open Geospatial Consortium (www.opengeospatial.org) or the World Wide Web Consortium (www.w3.org) for web applications. Important topics for standards in integrated modelling include temporal and spatial definitions, phenomenon dictionaries, metadata, time-stepping and interface definitions.

Developers around the world have written tools and applications that use these standards and, as a result, a wide range of software packages has become available to end-users, many of which are open source. Gregersen *et al.* (2007) maintain that a successful standard has three characteristics:

- (1) It is technically sound.
- (2) Adoption reaches a critical level.
- (3) The standard is supported and developed to meet new demands and cope with changes to utilized software packages.

This emphasizes the need for a community to develop around a standard, or software product, for it to become successful.

Application to integrated environmental modelling.

Khatibi *et al.* (2004) noted that the different areas where a paradigm is applied may be at different stages. Hydraulic modelling is in the norming stage, with the development of standard approaches. When the stages of systemic knowledge management are applied to the broader field of integrated environmental modelling, the following developments can be identified:

- **Pre-paradigm:** development of single models and modelling systems. The sequential running of different models with data exchange via output files.
- **Forming:** an appreciation developed that separate models can be linked at run-time (Reed *et al.* 1999; van der Wal & van Elswijk 2000; Havnø *et al.* 2001; Gijssbers *et al.* 2002; Whelan & Nicholson 2002). Individual projects set up integrated environmental modelling systems, such as HarmonIT (Gijssbers *et al.* 2002; Blind & Gregersen 2005; Moore & Tindall 2005). Different approaches covered the same ground or separate areas within the subject (Safiolea *et al.* 2011; Bastin *et al.* 2013; Lu & Piasecki 2012).
- **Proliferating:** Many approaches and systems were developed and tested. Bastin *et al.* (2013) divided existing frameworks for model coupling

into three classes: (i) standard languages and interfaces; (ii) workflow and integration tools; and (iii) frameworks and provided examples. Lu & Piasecki (2012) used the categories: (iv) geographical area; (v) monolithic super model; (vi) component-based modelling framework; and (vii) coupling framework. Examples from Bastin *et al.* (2013) and Lu & Piasecki (2012) are given in Table 4 and Table 5, respectively.

- **Norming:** The current situation appears to exhibit the onset of the norming stage as there is a clear appreciation of the need to adopt common approaches, which are often manifest as a formal standard. Many standards, candidate standards and implementations of standards have already been developed (see Tables 4 & 5 for a selection). We expect that a limited number of approaches and standards will flourish, covering a complementary set of required functions. The members will be decided by natural selection; practitioners will choose which approach to adopt, so some will flourish and some will become redundant. However, this process will be augmented by the belief that consolidation will be needed. The competitive advantage offered by an approach may depend on a number of things, including the availability of funding to develop and maintain the core software, the development of an active community, the ability to link to other software (for example, the approach can be used over the Web or within different workflow tools). An open architecture will be crucial and the use of open source software is likely to increase. This movement has been boosted by the recent switch of existing closed source codes into open source codes by major players in the hydraulic modelling community such as TELEMAC (www.opentelemac.org) and elements of Delft3D (www.deltaressystem.com). However, given the great resources devoted to legacy code, its track record and the number of instances of its successful use, the ability to include closed source code will remain an advantage for years to come.
- **Performing:** The performing stage will be achieved when high performance can be achieved within a flexible environment, where custom solutions can be delivered to meet clients' needs and the clients are aware of the limits of the system (Khatibi 2003a). Although the owners of individual environmental modelling systems may each claim to be at this stage now, the entire community encompasses too many alternative approaches without any single one offering all that is required. The performing stage will be reached after more testing, revision and consolidation of approaches.

FROM INTEGRATION TO FUSION

Table 4. *Elements in integrated environmental models (Bastin et al. 2013)*

(i) Standard languages and interfaces	
Business Process Execution Language	docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf
Open Modelling Interface	www.openmi.org
Common Component Architecture	www.cca-forum.org
Predictive Model Markup Language	www.dmg.org
Interactive Component Modelling System	www.clw.csiro.au/products/icms/
(ii) Workflow and orchestration tools	
Taverna	www.taverna.org.uk
Kepler	https://kepler-project.org/
VisTrails	www.vistrails.org
Trident	tridentworkflow.codeplex.com
(iii) Frameworks	
Delta Shell	Donchyts & Jagers (2010)
Model Coupling Toolkit	www.mcs.anl.gov/research/projects/mct/
Earth System Modelling Framework	www.earthsystemmodeling.org
Ocean Atmosphere Sea Ice Soil	www.cerfacs.fr/3-26568-OASIS.php
Community Earth System Model	www.cesm.ucar.edu
O-PALM	www.cerfacs.fr/globc/PALM_WEB/
Bespoke Framework Generator	Armstrong <i>et al.</i> (2009)
Tarsier	ecoviz.csumb.edu/wiki/index.php/Tarsier
FluidEarth	https://fluidearth.net
Integrated Component Modelling System	www.clw.csiro.au/products/icms/
The Invisible Modelling Environment	www.toolkit.net.au
Spatial Modelling Environment	Maxwell & Costanza (1997)
Framework for Risk Analysis of Multi-media Environmental Systems	mepas.pnnl.gov/framesv1/sum3ug.stm

All web links accessed 4 November 2014.

Tables 4 and 5 indicate that there are many different approaches and solutions available. All have restrictions on their use and are, to a variety of extents, tailored to the issues facing their user communities. The question of how this plethora of approaches might be consolidated will be addressed

Table 5. *Elements in integrated environmental models (Lu & Piasecki 2012)*

(iv) Geospatial context	
Chesapeake Community Modelling Programme	ches.communitymodeling.org
(v) Monolithic code framework	
Weather Research and Forecasting	www.wrf-model.org/index.php
(vi) Generic component-based modelling framework	
Community Surface Dynamics Modelling System	csdms.colorado.edu
Partnership for Research Infrastructures in Earth Systems Modelling	Valke <i>et al.</i> (2006)
Earth System Modelling Framework	www.earthsystemmodeling.org
(vii) Coupling frameworks	
Modular Modelling System	Leavesley <i>et al.</i> (1996)
Object Modelling System	David <i>et al.</i> (2004)
ModCom modular simulation system	Hillyer <i>et al.</i> (2003)
Community Hydrologic Modelling Platform	https://www.cuahsi.org/
Dynamic Information Architecture System	Campbell & Hummel (1998)
Tarsier	ecoviz.csumb.edu/wiki/index.php/Tarsier
Open Modelling Interface	www.openmi.org
Interactive Component Modelling System	www.clw.csiro.au/products/icms/
The Invisible Modelling Environment	www.toolkit.net.au
Spatial Modeling Environment	Maxwell & Costanza (1997)
Next-generation Framework for Aquatic Modelling of the Earth System	Lakhankar <i>et al.</i> (2008)

All web links accessed 4 November 2014.

later. First, the relationship between data and models is considered in light of the needs of integrated environmental modelling.

Data and models

Traditionally, the boundary between measured and modelled data has been presented as being quite clear: data are direct observations used to set up bathymetry and boundary conditions, and also used for verification and validation. Model runs are simulations that produce outputs (modelled data) that can be compared with other measured data.

However, all data are abstractions of reality; there are merely different levels of abstraction. Possibly the lowest level involves a direct observation of the environment (for example, water level up a marked rule). However, it is impossible to collect much data through direct observation and so instruments are used for repetitive observations. At the next level of abstraction, an instrument will utilize a well-understood mathematical relationship (or conceptual model) to derive the observational value (for example, water level based on pressure measurement). However, a much higher level of abstraction is required to calculate many derived quantities from observational data. For example, the calculation of suspended sediment concentrations from satellite data requires a modelling process, whereby algorithms, assumptions and calibration (normally using a different type of data) are all applied to the captured signal to provide 'measured' data. Hence, as data collection and analysis becomes more complex, the boundaries between model and data become blurred.

Moreover, there are many different types of numerical model, each of which is a representation of one part of reality (Cunge 2003). The following three model types are discussed here:

- deterministic numerical simulations
- data driven modelling
- data mining and assimilation.

Deterministic modelling

The traditional deterministic model is a numerical representation of a physical law or laws such as conservation of mass, energy or momentum. The equations are discretized and solved using a variety of numerical schemes. As such, they encapsulate our knowledge of the physics of a problem. However, some behavioural (or data-driven) representation of processes is needed, such as the use of a roughness length in shallow water flow modelling. This is even more evident in the modelling of more complex, less well-understood phenomena such as sediment transport models. These range from the

simulation of scour round an object using computational fluid dynamics code (Dixen *et al.* 2013) through coastal area modelling, to beach plan-shape modelling and models of the coastal tract. These models, even the most detailed, contain behavioural representations of sediment transport at one length-scale or another.

Data driven models

A data driven model is a means of deriving a functional relationship between input and output data, where the parameters and coefficients have been fitted to the data, and so are not based on physical laws (Cunge 2003). Data driven model types include correlations, autoregressive-moving-average (ARMA) methods, artificial neural networks, genetic algorithms and genetic programming. These models are rapid to run, but depend on the number, range and accuracy of the input data. Cunge (2003) warned against possible misuse of these models, but they have become increasingly popular in the past decade and commonly feature in almost any recent issue of popular journals such as the *Journal of Hydroinformatics*.

However, the application of such methods to environmental modelling will result in new transformations and relationships that have not been thought of, but arise out of the formalized exploration of large data sets and the power of recursive algorithms that are made possible by computer programs rather than the physical laws that would underpin a deterministic model. For example, it will be possible to treat deterministic models as data providers (or the sources of the variables to be transferred to another model) and use a data driven model to evolve a sensible overall scenario.

Data mining and data assimilation

Cunge (2003) argued that a theory is both a description and an explanation of physical processes, and that data driven and data mining approaches are therefore not theories (and, by implication, are less worthy than theories and their deterministic models). However, the idea of using computers to augment human intelligence dates back to the memex (Bush 1945), while today some areas of science such as astronomy and biology have collected so much data that data mining techniques are being used to extract information (in the form of statistical models of complex phenomena) that cannot be determined by human intelligence alone. These techniques are not generally used with environmental data (although the volumes of remote sensing data being collected are huge) but they cannot be ignored, even when accepting many of the caveats that Cunge (2003) supplies.

Data assimilation takes measured data and incorporates it into the running of a numerical modelling suite, influencing the final outcome. Data assimilation starts to integrate data and models explicitly and is an area of active research in many modelling disciplines.

Implications for integrated modelling

The increasingly blurred boundaries between measured and modelled data are one driver for the development of common standards for its interoperation. The standards for model linking should, as a corollary, allow for the import of data from more permanent storage media. File-based data transfer is sometimes inefficient but offers a well utilized and simple structure for data from a variety of sources targeted at applications for accessing, reading, writing and analysing.

Modern data standardization is tending to occur at two levels: the structure of the data and its technical implementation. Definitions of data structure are independent of the file encoding. For example, ISO 19115 outlines the data structure of spatial metadata with its XML encoding given in ISO 19139. The supporting (use and discovery) metadata can be given in separate files to the values themselves. This is exhibited in formats such as CSML, NetCDF and XDMF, which offer a binary file type (such as HDF5) for high volumes. Also, directives such as the one establishing an Infrastructure for Spatial Information in the European Community (INSPIRE) (<http://inspire.jrc.ec.europa.eu>) provide a legal and technical framework for data interoperability. INSPIRE includes specifications for the data, discovery, use and download services and is aimed at making the finding, using and sharing of data easier across the European Union (EU). However, for any practitioner wishing to offer a dataset to the wider community, the set of standards on offer is incomplete, overlapping and highly esoteric.

The Open Modelling Interface

The need for integrated environmental modelling tools led to the development of the Open Modelling Interface (OpenMI) during two European Commission funded projects, HarmonIT (2002–2005) and OpenMI LIFE (2006–2010). The particular driver for these projects was that whole catchment modelling is required to deliver the objectives of the Water Framework Directive (European Commission 2000) which requires Member States to achieve ‘good ecological status’ of surface waters by 2015. This places significant demands on water managers and requires the linking of a wide variety of models. Integrated modelling was seen

as a realistic mechanism for this, which would enable process interactions to be simulated across catchments (Gijssbers *et al.* 2002; Blind & Gregersen 2005; Moore & Tindall 2005). The creation of an open modelling environment was intended to capitalize on the huge prior investment in model development (which was mainly in proprietorial code). It was inspired by a number of national initiatives that demonstrated the feasibility of integrated modelling frameworks and involved work with major commercial players in the water resources software market to ensure that the vast amount of encapsulated knowledge in existing (proprietorial) tools was not abandoned, but rather was modernized, recycled and reused (Gijssbers *et al.* 2002; Blind & Gregersen 2005; Moore & Tindall 2005).

The chosen path to integrated modelling was the development of the OpenMI standard, a set of software interfaces that a compliant component must implement. Version 1.0 (.Net) was released at the end of HarmonIT (Gregersen *et al.* 2005, 2007). Implementation of this standard was tested for a wide range of cases in the OpenMI LIFE project under the European Commission’s LIFE Environment programme. OpenMI was applied in the Scheldt basin in Belgium and the Netherlands (Safiolea *et al.* 2011) and in the Pinios basin in Greece (Makropoulos *et al.* 2010; Safiolea *et al.* 2011) to demonstrate that OpenMI can assist competent water authorities in joint model integration to achieve the objectives of the Water Framework Directive. The standard was updated, as was the release procedure, leading to OpenMI version 1.4 (available for both .Net and Java), which became the only official version of the standard. In addition the project set up a legal body, the OpenMI Association, to support, maintain and publicize the OpenMI standard.

Development work continued and, by the end of OpenMI LIFE, a beta release of OpenMI version 2.0 was published for external review. Work continued and version 2.0 of the standard (OpenMI Association 2010a) and reference (OpenMI Association 2010b) were officially released in December 2010 during a EU–US summit in Washington DC. Following discussions with the Open Geospatial Consortium (OGC), OpenMI has become an OGC standard (<http://www.opengeospatial.org/standards>).

OpenMI standard

The stated aim in the development of OpenMI was to provide a mechanism for physical and socio-economic models to be linked to each other, other data sources and tools at run-time (Gijssbers *et al.* 2002; Blind & Gregersen 2005). This was achieved through the development of the OpenMI standard

(Gregersen *et al.* 2005, 2007; Gijsbers *et al.* 2010) which is a software component interface (cf. Khatibi *et al.* 2004) that enables OpenMI components to:

- be configured to exchange data during computation (at run-time);
- run simultaneously and share information at each time step making model integration feasible at the operational level.

The OpenMI standard was originally conceived to facilitate the numerical modelling of interacting environmental processes related to whole catchment modelling. However, what was developed is a generic solution to the problem of data exchange between models or software components. For example, it can be applied to link models of different domains and environments such as models of hydraulics, hydrology, ecology, water quality and economics. It can be used to link models of different dimensionality, so that a one-dimensional (1D) river model can be coupled to a two-dimensional (2D) flow model when the river broadens or a 2D flow model could be coupled to a three-dimensional (3D) flow model. OpenMI can also be applied to link models that operate at different time steps, so are running asynchronously. It can link different spatial representations (e.g. networks, grids, polygons) and can cope with different projections, units and categorizations, and with models that have no temporal or spatial representation.

Examples of the use of OpenMI include, but are far from limited to, the following:

- Becker & Schüttrumpf (2011) used the programming interface of the finite element groundwater flow model Feflow to achieve OpenMI compliance without altering the source code by implementing remote procedure calls. This was demonstrated using a model of dike seepage under transient water level boundary conditions.
- Bulatewicz *et al.* (2010) demonstrated the integration of agriculture, groundwater and economic models using OpenMI, while Bulatewicz *et al.* (2013) wrote a Simple Script Wrapper, which simplifies the linking of scripted models (in Matlab, Scilab or python, for example) to other OpenMI components.
- Shrestha *et al.* (2012) linked models of hydrology, hydraulics, water temperature, sediment transport and faecal bacteria using OpenMI and applied this composition to the River Zenne in Belgium, demonstrating interaction between the sediments and the bacteria.
- Becker *et al.* (2012) linked the software package RTC Tools (for the control of hydraulic structures) to the hydraulic model SOBEK using OpenMI to provide a real time control package for hydraulic structures on the River Rhine.

OpenMI compliant components may come from any suppliers and can be based on legacy software or a new model. The standard supports two-way links (Gregersen *et al.* 2007) where involved models mutually depend on calculation results from each other. Castronova & Goodall (2013) found that computational overhead imposed by OpenMI's run-time exchange of data was not significant when applied on a semi-distributed watershed model. Developers have also found it practical to use OpenMI in conjunction with other software tools, as listed in Table 6.

The new features in OpenMI 2.0 centre on making the standard more flexible and extensible. They include the following.

- (1) The concept of adaptors which allow component outputs to be transformed (adapted) before inputting into other components. This is to allow situations such as transformation between differing spatial structures held by different components (for example: a triangular model grid passing data to a rectangular grid; and a 2D grid passing data to 1D). It is also possible to chain adaptors together if multiple transformations are required in series.
- (2) Easier incorporation of data from other sources such as files, databases and web services.
- (3) A more flexible overall structure with a core set of mandatory interfaces and optional extension sets. The extension governing space- and time-dependent components is included in the current edition as this is the most common, current requirement, but is not part of the mandatory interface set. This allows the core standard to be easily applied by other model types.
- (4) A representation of geographical data structures which is closer to common, modern implementations. This points towards OpenMI dovetailing with specific geospatial data standards in the future.

Roles and responsibilities

The OpenMI standard allows the passing of many different types of data between models, which is a great strength. The more widely applicable an interface is, the closer it gets to 'plug and play' interoperability. However, this is also a weakness, as when the description of the passed data becomes less prescriptive, the onus passes to the modeller to understand what is being offered by one model and required by another, as this is not specified by the standard.

There is already an emerging issue with large complex models in that many assumptions used in the construction of suitable algorithms and code

Table 6. *Software tools that have been linked to OpenMI*

Software linked to	Reference
Simple Script Wrapper	Bulatewicz <i>et al.</i> (2013)
CUAHSI Hydrologic Information System (HIS)	Castronova <i>et al.</i> (2013)
Simple Model Wrapper	Castronova & Goodall (2010)
Real Time Control (RTC) tools	Becker <i>et al.</i> (2012)
UncertWeb	Gupta <i>et al.</i> (2012), Bastin <i>et al.</i> (2013)
Pyxis workflow manager	San Roman Blanco <i>et al.</i> (2012)

are not explicit. When the user was also the developer, an understanding of these assumptions was retained. Increasingly users do not have this knowledge and model developers have been slow to help make the underlying assumptions explicit, for example, by incorporating tests that check whether an assumption is being violated and warning the user. As we begin to link models that deploy different sets of assumptions this problem is compounded.

So, for example, consider the scenario where there are two models passing data between themselves during run-time (see Fig. 2). A person constructing a composition by linking the models needs to know:

- the assumptions that underpin each model, so their suitability for the target application can be judged;
- that both models are OpenMI compliant (or how to achieve this);
- that they can obtain the model under suitable licence terms;
- details of the data being offered by the first model;
- details of the data expected by the second model;
- how to adapt the output from the source model to suit the target.

It is not always clear where this information will come from. Ideally a model should be developed with the following accompanying elements:

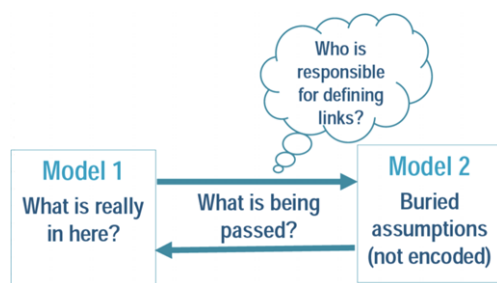
- strong version control (especially if open source);

- documentation and training material;
- an application for verification and validation (ideally one that is linked to version control, such as that by Farrell *et al.* 2011);
- improved ‘use’ metadata describing the model itself, in particular the underlying assumptions and interfaces;
- a track record including examples of use;
- increased compliance to recognized versions of standards;
- a standard licence;
- supporting web portals with standardized cataloguing and documentation.

The scientists who write component models are not normally software developers familiar with standards and may not themselves have any need to link their models. Neither scientist nor software developer has the wrapping of models as a primary goal, so the wrapping of models falls between two stools (Knapen *et al.* 2013). This situation could be assisted by the provision of software tools and guidance to simplify the model wrapping process.

Moreover, unless the model wrapping is documented along with the underlying component model, it may not be clear which variables are exposed (available to be exchanged) or exactly how they are defined. As an example of the latter, radiation stress is defined differently in the wave model SWAN (Booij *et al.* 1999) and the shallow water flow model TELEMAC-2D (Hervouet 2007) so when they are linked in an OpenMI composition, an adaptor must be used to translate the output from SWAN for TELEMAC-2D.

To address the challenges of integrated modelling, the community will have to start to provide more information about their models, as suggested above. It will only be by providing this information, ideally in a standardized way, which would be assisted by the development of common ontologies, that people will trust models enough to take up and use third-party models to create new model compositions. Moreover, the community will have to address questions such as ‘who owns the intellectual property rights to a new composition?’ and ‘what is the quality of the modelled data?’.

**Fig. 2.** The perils of plug and play.

FluidEarth

FluidEarth, formerly known as OpenWEB, is a collaborative initiative between the academic community and users with the aim of researching and implementing integrated computer modelling approaches to environmental systems (Pearce *et al.* 2010). One of the main problems with the OpenMI standard is that it requires application development skills above those of a typical scientific programmer. In response, HR Wallingford has developed the FluidEarth implementation of the OpenMI standard (Harpham *et al.* 2014) making its use easier for the scientific community through the provision of a graphical user interface, 'Pipistrelle', and a software development kit. Both are open source and available on SourceForge (<http://sourceforge.net>).

FluidEarth is developing a dialogue between key academic partners, providing the tools needed to reduce duplication of effort within the research community, making the task of translating research into applications easier and increasing the commercial potential of research outputs by creating a large community of active users. To facilitate these activities, FluidEarth has developed a web portal (<http://fluidearth.net>) where people can find out about the FluidEarth implementation of the OpenMI standard, post discussion questions and replies, present their studies, view e-learning tutorials, read community announcements and

gain access to the following key software and repositories:

- (1) Pipistrelle, the FluidEarth open source user interface, provides a run-time environment for linking OpenMI compliant components. This gives modellers the ability to create and run compositions of linked components (Fig. 3).
- (2) The FluidEarth Software Development Kit (SDK) is an open source tool allowing model developers to more easily adapt their models and other components to be compliant with the OpenMI standard, thus reducing the requirement for specialist programming.
- (3) A repository is provided for research partners to upload software that can be tried and tested by the community.
- (4) The FluidEarth catalogue of model engines and instances of their application is an implementation of the open source GeoNetwork web interface for searching geospatial data (<http://geonetwork-opensource.org>).

A SourceForge project (<http://sourceforge.net/projects/fluidearth>) gives access to all source code, feature requests and amendments of the FluidEarth 2.0 tools for version 2.0 of OpenMI.

FluidEarth communities

FluidEarth recognizes the split in responsibilities and skills that has occurred in the development of

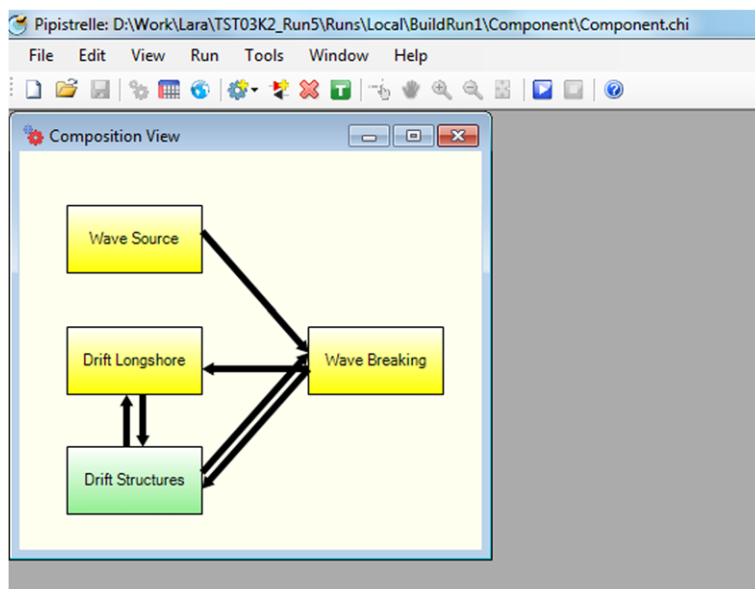


Fig. 3. Screenshot of Pipistrelle editor with simple composition.

FROM INTEGRATION TO FUSION

modelling tools (from Abbott's first to fifth generations and from Khatibi's forming to performing stages) and seeks to facilitate the activities of the following five main groups and to act as a conduit between them.

- (1) **Software architects and developers** who develop and test software tools for common tasks. They have a strong background in software development, but may have no expertise in environmental modelling.
- (2) **Model integrators**, whose role is to take individual models and form integrated models (or compositions) from them. The role of model integrator is a new one and requires a particular set of skills in using integrating software (such as the FluidEarth SDK and Pipistrelle) and checking the metadata to ensure compatibility.
- (3) **Researchers** wishing to develop new techniques for integrated modelling to explore more complex feedback mechanisms, or simply test new algorithms within existing models. They are commonly scientists with a knowledge of coding but with limited knowledge of accepted software development practices, who are used to working on the development of individual models.
- (4) **Users** who apply existing models, or even model compositions, to real world problems. They are likely to be scientists or engineers, with a background in environmental or physical sciences or engineering.

- (5) **End-users**, such as the developers of schemes, regulatory authorities, local and national governments who use the results supplied by users to influence their decision making (for example in developing policy or planning).

Members of the FluidEarth community may be in more than one group. For example, it is not unknown for a researcher to contribute to software development as well as the writing of individual models. It is more common perhaps for a user to also be a researcher. End users may well not have any background in science or engineering, or any experience of coding models or software. The groups are interdependent and share the FluidEarth resources, as shown in Figure 4.

Towards model fusion

Previous sections have shown how the demand for modelling across disciplines has led to the development of a large number of solutions to the problem of achieving integrated modelling, which may lead ultimately to model fusion. As noted in the introduction, fusion involves the 'melting [or] blending of different things into one' (Oxford English Dictionary), so it involves not just the ability to link models, but also easy access to information about the models and linking technologies and access to software tools to make the process easier.

Using Khatibi (2003a), we propose that model fusion will occur: when problems can be addressed

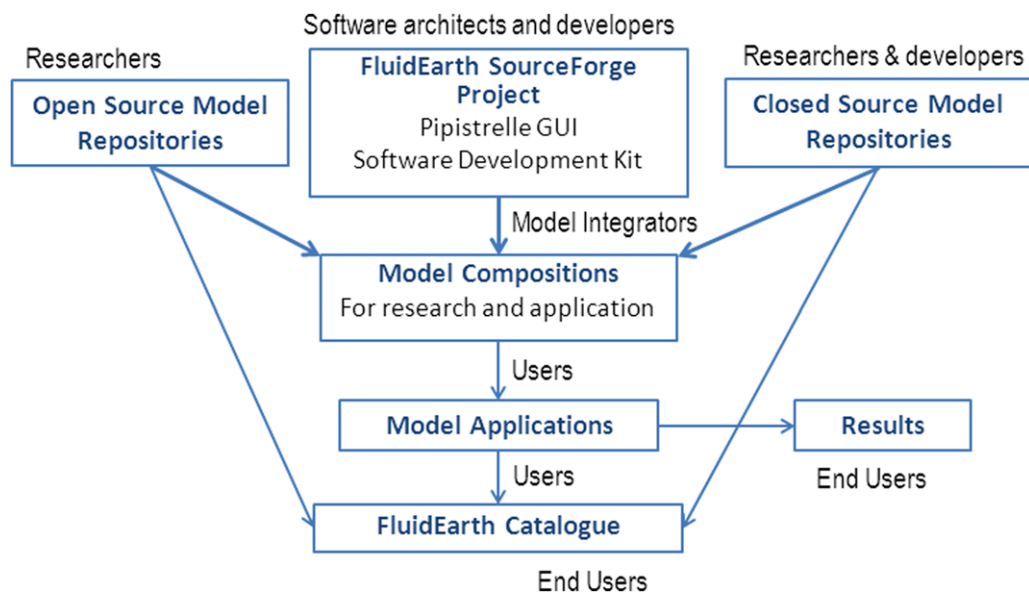


Fig. 4. FluidEarth#software, repositories and communities.

efficiently within a flexible environment; where custom solutions can be delivered to meet clients' needs and the clients are aware of the limits of the system; and where the connections between components and the influences they have on each other are understood and utilized. There is an interplay between technical, economic and social needs. Model fusion will occur when integrated environmental modelling has reached the performing stage (Khatibi 2003a).

Many approaches to integrated modelling already exist (Tables 4 & 5) but the norming phase will involve conscious attempts at consolidation, as part of a Darwinian struggle for survival, and the emergence of a few key sets of standards. Note that it is highly improbable that a single approach will evolve from this process, as no approach is likely to be optimal for all situations. It is much more probable that a reduced number of approaches will co-exist, some in niche markets and some in open competition. This will help to drive continued innovation.

We put forward in this paper an approach to achieving model fusion, not just considering technical issues but also the involvement of the communities of model developers, scientists, users and end-users which develop around an integration method. This approach recognizes the emergence of an era of more open science. The concept of integrated environmental modelling benefits from open architecture and is commonly associated with open source software. Meanwhile the importance of open access to public sector research has been recognized by the Organisation for Economic Co-operation and Development (OECD 2004) the European Commission (European Commission 2011) and the UK (HM Government 2012). The European Commission has launched an Open Data Strategy for Europe, which is expected to deliver a €40 billion boost to the EU's economy each year (European Commission 2011) and there is a commitment to increasing access to data and publications within the Commission's Horizon2020 research programme. Open access to research publications is becoming increasingly common and open access journals are proliferating. Moreover, we are in an era of increasingly open scientific collaboration. Nielsen (2011) presents a number of case studies of networked science, from citizen science to mathematical problem solving, which have occurred by open collaboration over the Web and advocates this as the future of science.

The move towards open science, comprising open architecture, open source, open data, open access publication and open collaboration (or networked science) is ongoing but faces many difficulties. Nielsen (2011) predicts that it will only take place when we learn to value openness and the

sharing of models and data as much as our publication and citation records. However, openness will only work efficiently through the widespread application of standards, as these assist people in working together. This paper is not concerned with how to make open science happen, but how to move integrated modelling towards model fusion against this backdrop.

We have seen in this paper the importance of standards, software architecture, model metadata and documentation, and tools to assist with linking models and collaboration. These ideas form the basis for the following six topics:

- (1) Enhanced metadata for data and models
- (2) Provision of supporting information
- (3) Software-as-a-service
- (4) Consolidation of linking technologies
- (5) Diagnostic and reasoning tools
- (6) Verification, validation and explanation

Each of the six topics is described in more details below.

Enhanced metadata for data and models

There will be benefits to enhancing the richness of information about shared data and models and in standard forms. This extra information will enable any interested party to judge the suitability and quality of the data or model. In order to do this we need to develop (and ideally consolidate) metadata standards and ontologies. A primary requirement will be to extend from an established standard set to reduce the likelihood of independent bespoke implementations of the same standard being developed. Scientists and engineers will have to get used to the routine generation and use of metadata. Tools that map between different standard ontologies are beginning to arrive in the marketplace and will become increasingly useful.

We will also need to improve the standards of model documentation. There are already forms of automatic code documentation, such as Doxygen (www.doxygen.nl/features.html) which show flow charts of information between sub-routines, but these are themselves generally not sufficient to understand a code. A poorly documented open source code is a considerable barrier to understanding and hence take-up, while a model without documented verification or validation is impossible to judge.

A community of modellers will also be more effective when there are improved tools to search for, discover and link to data and models. Indeed, tools that address many semantic issues between communities will become more fundamental as integrated modelling increasingly crosses disciplines.

Provision of supporting information

The development of shared knowledge bases will help to create a free market in information, which every member of the community can access equally. This is more manageable within an organization than between organizations, but the concept is similar to a distributed database, where the data owner maintains the data, with the ability for these data to be integrated with other data held in the distributed system. This will require protocols and standards to be developed, published, accepted and used.

Harvesting data or models from external knowledge bases is a much bigger challenge, not so much due to the technical problem of searching for information over the Web or having common formats, but due to the need to establish ownership, licence conditions and the quality of both data and models; hence the need for enhanced richness of information about data and models. The knowledge bases also have to contain information on input/output protocols, the principles underpinning each model, implicit assumptions in the models, and information on data calibration and processing.

Software-as-a-service

The evolution of modelling systems and the changing role played by the user (Table 2) combined with the changes in the computing environment (Table 3) indicate a move towards offering Software-as-a-service (SaaS) over the internet. Although this is common in other fields and has been trialled in academic circles, it is only starting to be offered as a commercial proposition in environmental modelling (Bourban *et al.* 2012). This has been enabled by rapid progress in the development of the Cloud, the Grid and emerging standards for web services.

Consolidation of linking technologies

There are a considerable number of candidate approaches to model integration (Tables 4 & 5). These come with significant overlap in functionality, but also many distinct features which usually arise from the specific needs of the community that created them. Some consolidation is sensible and inevitable as the community comes to fully appreciate the benefits to be gained from adopting common standards. The particular needs of each community and the degree of overlap will severely hamper this process and, of course, nobody wants to abandon the time and effort they have put into developing their own system. Moreover, anyone changing systems will have to make an investment of time and effort in learning the new system and

there are limited incentives to make this happen. In the short term, consolidation may occur as researchers follow the money and/or join in with active communities. In the longer term, we propose that one or both of two outcomes are possible:

- (1) A set of mutually compatible standards will be adopted to solve particular issues, such as standards for metadata (e.g. ISO 19115) web services (e.g. WPS) and memory-based model coupling (e.g. OpenMI) file-based model coupling (with a variety of standard file formats). These standards will be universally adopted by virtue of their utility and technical credibility. Existing frameworks will adapt to incorporate these standards into the appropriate components.
- (2) A large technology company will produce a product which covers a large percentage of modelling requirements. Irrespective of the quality of the underlying product, it becomes 'standard' because it exists, is common and works with minimal technical help.

Diagnostic and reasoning tools

The development of integrated environmental models, composed of a number of component models (each of which may have many sub-routines) calls for the development of a new set of tools for testing integrated models, analysing the results and synthesizing outputs.

Each model should have its own published verification and validation tests, preferably including measures of model skill (Sutherland *et al.* 2004) rather than just qualitative assessments. Models should be developed under version control and there should be regular, preferably automated (Farrell *et al.* 2011) testing of new code, ranging from unit tests of functions and sub-routines to validation of the entire model.

However, we also need to test integrated models to ensure that the data exchange and feedback mechanisms have been adequately captured. This should also involve skill scores, if at all possible, and it may be possible to adapt techniques such as variance based sensitivity analysis (VBSA) (Saltelli *et al.* 2008) to isolate which model parameters contribute the most to the variance in the output result.

As multidisciplinary model compositions are often time-consuming to run, the use of model emulators may be developed and the mixed use of a full model composition and a quicker emulator has the potential to speed up VBSA or optimization techniques with integrated models. Where a process or link is poorly understood, a simple behavioural representation may be developed, with the coefficients or even the form of the behavioural representation optimized using data-driven techniques. Another

concept involves replacing a poorly defined or uncertain model link with data in some simulations and either comparing the results from the full model and the model/data mix, or allowing the linked model data to influence but not entirely define the data passed (Voinov & Cerco 2010).

The development of diagnostic and reasoning tools involves bringing in the tools of professional software development and is already changing the skill sets needed to develop software. The modelling community will continue to need people with a computational sciences background as well as people with a background in applied sciences.

Verification, validation and understanding

Models are only useful if they are used in a way that acknowledges their limitations and serves to help establish and inform understanding. There is always a danger that with increasing complexity, clarity of understanding is lost and users become confused under the welter of, often conflicting, information being provided. What is worse, models, like statistics, can be used to obfuscate or mislead. This is already giving rise to public scepticism about the use and value of models. Too often model outputs have been 'over sold', or conversely have been shown to have missed critical outcomes (for example, the infamous 'hurricane' that was not going to hit the south of England in 1987 (Met Office 2013)). As a result there is a preference among non-specialists to prefer the simple over the complex and this can lead to 'rules of thumb' being preferred over sophisticated model results (no matter how good the representation of the physics, chemistry and biology). Given the inherent uncertainty that abounds in environmental modelling, results have to be communicated with care. A careful balance is needed between not undermining the value of the model outputs while at the same time recognizing the limitations of the model. Often the biggest gains are when a model, or models, leads to a better understanding of how a system behaves, rather than specific predictions.

This complexity also reinforces the need for better audit trails of the modelling that has been undertaken. All the various steps in the modelling process need to be documented, ideally in an openly accessible form that allows the whole process to be reproduced by others. This is rarely possible at present but the move to greater open access to data and models should enable the possible for in-depth and independent reviews of model applications. However, there will also be a need for improved interrogation tools that allow the model user to generate reports on particular aspects of the model. This is not dissimilar to the process of querying a complex database. The requirement is

to be able to drill down from high level, perhaps summary outputs, to the underlying information. This is likely to include detailed outputs, data inputs, resolution in space and time, assumptions made in each of the component models, details of the exchanges between models, and the underlying science that is being represented in the model. Whilst metadata together with model documentation and manuals are a good start, they are a long way short of providing what is needed for a fully interactive set of interrogation tools.

Discussion

Many models are already very large, with hundreds of sub-routines and a host of assumptions buried deep in the formulation and coding of the model. Individually they can be a challenge to maintain. If such models are merged, maintenance becomes even more difficult. One of the biggest advantages of fusion through linking is that individual model developers retain responsibility for their model, while at the same time enabling others to use the model with other models to emulate the 'larger' model. This is not dissimilar to the evolution of databases. Initially it was thought that a central database was essential for the efficient use of data in an organization. After a few years it quickly became apparent that such databases were extremely difficult to maintain and that it was far better to have a distributed system, where the data owner maintained the data, with the ability for these data to be integrated with other data held in the distributed system.

This comparison with databases is also relevant in the context of the future development of model fusion. As models increasingly use data assimilation techniques and data monitoring uses models to aid interpolation, the difference between model and data becomes increasingly blurred. We are moving towards a new era, in a world that is data rich and with the ability to undertake large computational activities. This creates new opportunities for modellers in terms of the model complexity, model detail, and perhaps most importantly, the ability to run model ensembles and so start to better represent the uncertainty inherent in the analysis of real world systems.

However, this comes at a price. The new role of model integrator requires a different set of skills from that of researcher or user (although in practice they may be the same person) in order to ensure the validity of what is being represented by the collection of models. This requires a much clearer appreciation of the fact that all models are wrong but some are useful (Box 1979), not only amongst modellers but also those who use the outputs.

FROM INTEGRATION TO FUSION

Looking further ahead one might envisage the ability for models to ‘self-assemble’ compositions to represent a particular problem. This requires some substantial advances from where we are now, not least making explicit the knowledge and assumptions that are currently buried deep in existing models.

Conclusion

The past decade has seen the development of the new field of integrated environmental modelling where compositions of linked models exchange data at run-time. The application of systemic knowledge management to integrated environmental modelling indicates that we are at the onset of the norming stage, where gains will be made from the hierarchical organization of the competing options. This implies that there will be consolidation in the range of approaches that have proliferated in recent years, which is likely to become manifest in the predominance of a limited number of standards (covering ontologies, metadata, model interfaces, data formats and so on). An open software architecture (consisting of a user interface with published interfaces to a range of models, data and data processing routines) will be crucial and the use of open source software is likely to increase.

We propose six topics that will help integrated modelling to move through the norming stage towards the performing stage, where problems can be addressed efficiently within a flexible environment and custom solutions can be delivered to meet customers’ needs. The six topics are:

- (1) Enhancing metadata for data and models to enable any interested party to find and then judge the suitability and quality of the data or model.
- (2) Provision of supporting information to create a free market in information including the ownership of models, their licence conditions, underlying principles, input/output protocols, calibration, verification and validation.
- (3) Software-as-a-service, where models are offered for use through a web-interface.
- (4) Consolidation of linking technologies, such as OpenMI, where one or both of two outcomes are possible: (i) a set of mutually compatible standards will be adopted to solve particular issues; and/or (ii) a large technology company will produce a commonly-available product that covers most needs and works with minimal technical help.
- (5) Development of diagnostic or reasoning tools for testing integrated models, analysing the outputs and synthesizing the results.

- (6) Verification, validation and understanding of integrated modelling should improve. Results have to be communicated with care, due to the inherent uncertainty in integrated modelling. Ways must be developed to acknowledge limitations and to help establish and inform understanding. This will require improved audit trails, tools for the propagation and assessment of uncertainty, and improved tools for the interrogation of integrated models.

When these six topics are developed, integrated modelling will have developed into model fusion, which involves the ability to link models, but also easy access to information about the models and interface standards (such as OpenMI) and access to software tools to make the process easier. In order for this to happen, a community must develop that is prepared to openly share information about models and data. This will be assisted by the adoption of standards, but will require the community to value openness and the sharing of models and data as much as it loves its publication and citation records. The development of a more open culture and the adoption of standards therefore go hand-in-hand.

This work was funded by HR Wallingford’s internal R & D programme as part of the FluidEarth initiative, by the *iCOAST* – *integrating coastal sediment systems* project, which is funded by UK Natural Environment Research Council under grant NE/J00541X/1 with support from the Environment Agency and by the European Commission’s 7th Framework Programme through the DRIHM project (grant number 283568) and DRIHM2US project (grant number 313122). The topics for model fusion were influenced by a presentation given by Professor Rizolli at the final meeting of the OpenMI-LIFE project.

References

- ABBOTT, M. B. 1991. *Hydroinformatics: Information Technology and the Aquatic Environment*. Avebury Technical, Aldershot, UK.
- ABBOTT, M. B. & VOJINOVIC, Z. 2009. Applications of numerical modelling in hydroinformatics. *Journal of Hydroinformatics*, **11**, 308–319.
- ABBOTT, M. B., TUMWESIGYE, B. E. & VOJINOVIC, Z. 2006. The fifth generation of modelling in hydroinformatics. In: GOURBESVILLE, P., CUNGE, J., GUINOT, V. & LIONG, S.-Y. (eds) *Proceedings of the 7th International Conference on Hydroinformatics*. Research Publishing, Singapore, 2091–2098.
- ARMSTRONG, C. W., FORD, R. W. & RILEY, G. D. 2009. Coupling integrated Earth system model components with BFG2. *Concurrency and Computation: Practice and Experience*, **21**, 767–791.
- BASTIN, L., CORNFORD, D. ET AL. 2013. Managing uncertainty in integrated environmental modelling: the UncertWeb framework. *Environmental Modelling &*

- Software*, **39**, 116–134, <http://dx.doi.org/10.1016/j.envsoft.2012.02.008>
- BECKER, B. P. & SCHÜTTRUMPF, H. 2011. An OpenMI module for the groundwater flow simulation programme Feflow. *Journal of Hydroinformatics*, **13**, 1–12.
- BECKER, B. P. J., SCHWANENBERG, D., SCHRUFF, T. & HATZ, M. 2012. Conjunctive real time control and hydrodynamic modelling in application to Rhine river. *10th International Conference on Hydroinformatics, HIC 2012*, Hamburg, Germany.
- BLIND, M. & GREGENSEN, J. B. 2005. Towards an Open Modelling Interface (OpenMI) the HarmonIT project. *Advances in Geosciences*, **4**, 69–74.
- BOOI, N., RIS, N. R. & HOLTHUIJSEN, L. H. 1999. A third-generation wave model for coastal regions, 1, model description and validation. *Journal of Geophysical Research*, **104**, 7649–7666.
- BOURBAN, S., DURAND, N., TURNBULL, M., WILSON, S. & CHEESMAN, S. 2012. Coastal shelf model of northern European waters to inform tidal power industry decisions. In: BOURBAN, S., DURAND, N. & HERVOUET, J.-M. (eds) *Proceedings of XIXth TELEMASCARET User Conference*, 18–19 October 2012, Oxford, HR Wallingford, Wallingford, UK, 143–150.
- BOX, G. E. P. 1979. Robustness in the strategy of scientific model building. In: LAUNER, R. L. & WILKINSON, G. N. (eds) *Robustness in Statistics: Proceedings of a Workshop*, 11–12 April 1978, Durham, NC, USA, Academic Press, New York, 201–236.
- BULATEWICZ, T., YANG, X., PETERSON, J. M., STAGGENBORG, S., WELCH, S. M. & STEWARD, D. R. 2010. Accessible integration of agriculture, groundwater, and economic models using the Open Modeling Interface (OpenMI): methodology and initial results. *Hydrology and Earth System Sciences*, **14**, 521–534.
- BULATEWICZ, T., ALLEN, A., PETERSON, J. M., STAGGENBORG, S., WELCH, S. M. & STEWARD, D. R. 2013. The simple script wrapper for OpenMI: enabling interdisciplinary modeling studies. *Environmental Modelling & Software*, **39**, 283–294, <http://dx.doi.org/10.1016/j.envsoft.2012.07.006>
- BUSH, V. 1945. As we may think. The Atlantic. <http://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/> [accessed 4 November 2014]
- CAMPBELL, A. P. & HUMMEL, J. R. 1998. The dynamic information architecture system: an advanced simulation framework for military and civilian applications. In: *Proceedings of the Advanced Simulation Technologies Conference*, Boston, MA, USA.
- CASTRONOVA, A. M. & GOODALL, J. L. 2010. A generic approach for developing process-level hydrologic modeling components. *Environmental Modelling & Software*, **25**, 819–825, <http://dx.doi.org/10.1016/j.envsoft.2010.01.003>
- CASTRONOVA, A. M. & GOODALL, J. L. 2013. Simulating watersheds using loosely integrated model components: evaluation of computational scaling using OpenMI. *Environmental Modelling & Software*, **39**, 304–313, <http://dx.doi.org/10.1016/j.envsoft.2012.01.020>
- CASTRONOVA, A. M., GOODALL, J. L. & ERCAN, M. B. 2013. Integrated modeling within a Hydrologic Information System: an OpenMI based approach. *Environmental Modelling & Software*, **39**, 263–273, <http://dx.doi.org/10.1016/j.envsoft.2012.02.011>
- CHORLEY, R. J. 1962. *Geomorphology and General Systems Theory*. United States Geological Survey Professional Paper 500B, 10p.
- CUNGE, J. A. 2003. On data and models. *Journal of Hydroinformatics*, **5**, 75–98.
- DAVID, O., SCHNEIDER, I. W. & LEAVESLEY, G. H. 2004. Metadata and modelling frameworks: the object modelling system example. In: *Transactions of the 2nd Biennial Meeting of the International Environmental Modelling and Software Society*, 14–17 June 2004, Osnabruck, Germany, International Environmental Modelling and Software Society, Manno, Switzerland, 439–443.
- DIXEN, M., SUMER, B. M. & FREDSE, J. 2013. Numerical and experimental investigation of flow and scour around a half-buried sphere. *Coastal Engineering*, **73**, 84–105. <http://dx.doi.org/10.1016/j.coastaleng.2012.10.006>
- DONCHYTS, G. & JAGERS, B. 2010. DeltaShell – an open modelling environment. In: SWAYNE, D. A., YANG, W., VOINOV, A. A., RIZZOLI, A. & FILATOVA, T. (eds) *IEMSS 5th International Congress on Environmental Modelling and Software*, 5–8 July 2010, Ottawa, Canada, International Environmental Modelling and Software Society, Manno, Switzerland, 1058–1065.
- EUROPEAN COMMISSION 2000. Directive 2000/60/EC of the European Parliament and of the Council of 23 October 2000 establishing a framework for Community action in the field of water policy. *Official Journal of the European Communities L*, **327**, 1–170.
- EUROPEAN COMMISSION 2011. Open Data, an engine for innovation, growth and transparent governance. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, COM(2011) 882 final, European Commission, Brussels.
- FARRELL, P. E., PIGGOTT, M. D., GORMAN, G. J., HAM, D. A., WILSON, C. R. & BOND, T. M. 2011. Automated continuous verification for numerical simulation. *Geoscientific Model Development*, **4**, 435–449, <http://dx.doi.org/10.5194/gmd-4-435-2011>
- GIJSBERS, P. J. A., MOORE, R. V. & TINDALL, C. I. 2002. HarmonIT: towards OMI, an Open Modelling Interface and environment to harmonise European developments in water related simulation software. In: FALCONER, R. A., LIN, B., HARRIS, E. L., WILSON, C., CLUCKIE, I. D., HAN, D., DAVIS, J. P. & HESLOP, S. (eds) *Hydroinformatics 2002. Proceedings of the 5th International Conference on Hydroinformatics*, Cardiff, UK. IWA Publishing, London, 1628–1275.
- GIJSBERS, P., HUMMEL, S. ET AL. 2010. From OpenMI 1.4 to 2.0. In: SWAYNE, D. A., YANG, W., VOINOV, A. A., RIZZOLI, A. & FILATOVA, T. (eds) *IEMSS 5th International Congress on Environmental Modelling and Software*, 5–8 July 2010, Ottawa, Canada, International Environmental Modelling and Software Society, Manno, Switzerland, 1081–1088.
- GREGENSEN, J. B., GIJSBERS, P. J. A., WESTEN, S. J. P. & BLIND, M. 2005. OpenMI: the essential concepts and

FROM INTEGRATION TO FUSION

- their implications for legacy software. *Advances in Geosciences*, **4**, 37–44.
- GREGERSEN, J. P., GJISBERG, P. J. A. & WESTEN, S. J. P. 2007. OpenMI: open modellingInterface. *Journal of Hydroinformatics*, **9**, 175–191, <http://dx.doi.org/10.2166/hydro.2007.023>
- GUPTA, T., JONES, R., BASTIN, L. & CORNFORD, D. 2012. Integrating OpenMI and UncertWeb: managing uncertainty in OpenMI models. In: SEPPELT, R., VOINOV, A. A., LANGE, S. & BANKAMPA, D. (eds) *Managing Resources of a Limited Planet: Pathways and Visions under Uncertainty*, IEMSS Sixth International Congress on Environmental Modelling and Software, 1–5 July 2012, Leipzig, Germany, International Environmental Modelling and Software Society, Manno, Switzerland, 1151–1158.
- HARPHAM, Q., CLEVERLEY, P. & KELLY, D. 2014. The FluidEarth 2 implementation of OpenMI 2.0. *Journal of Hydroinformatics*, **16**, 890–906, <http://dx.doi.org/10.2166/hydro.2013.190>
- HARVEY, H. & HAN, D. 2002. The relevance of open source to hydroinformatics. *Journal of Hydroinformatics*, **4**, 219–234.
- HAVNØ, K., SØRENSEN, H. R. & GREGERSEN, J. B. 2001. Integrated water resources modelling and object oriented code architecture. In: *Proceedings of Water Resources Modelling and Management*. Japan Academic Society of Hydraulics. Chuo University, Japan.
- HERVOUET, J.-M. 2007. *Hydrodynamics of Free Surface Flows: Modelling with the Finite Element Method*. John Wiley & Sons, Chichester, UK.
- HILLYER, C., BOLTE, J. & LAMAKER, A. 2003. The ModCom modular simulation system. *European Journal of Agronomy*, **18**, 333–343.
- HM GOVERNMENT 2012. *Open Data White Paper. Unleashing the Potential*. The Stationary Office, Norwich, UK.
- JANJIC, Z., GALL, R. & PYLE, M. E. 2010. *Scientific documentation for the NMM solver*. NCAR Technical Note NCAR/TN-477 + STR. National Center for Atmospheric Research, Boulder, CO, USA.
- KHATIBI, R. H. 2001. Some issues in computational hydraulic practices. *Journal of Hydraulic Engineering*, **127**, 438–442.
- KHATIBI, R. H. 2003a. Systematic knowledge management in hydraulic systems: I, a postulate on paradigm shifts as a methodological tool. *Journal of Hydroinformatics*, **5**, 127–140.
- KHATIBI, R. H. 2003b. Systematic knowledge management in hydraulic systems: II, application to hydraulic systems. *Journal of Hydroinformatics*, **5**, 141–153.
- KHATIBI, R., JACKSON, D., CURTIN, J., WHITLOW, C., VERWEY, A. & SAMUELS, P. 2004. Vision statement on open architecture for hydraulic modelling software tools. *Journal of Hydroinformatics*, **6**, 57–74.
- KNAPEN, R., JANSSEN, S., ROOSENSCHOON, O., VERWEIJ, P., DE WINTER, W., UITERWIJK, M. & WIEN, J.-E. 2013. Evaluating OpenMI as a model integration platform across disciplines. *Environmental Modelling & Software*, **39**, 274–282, <http://dx.doi.org/10.1016/j.envsoft.2012.06.011>
- KUHN, T. S. 1962. *The Structure of Scientific Revolution*. 3rd edn. University of Chicago Press, Chicago.
- LAKHANKAR, T., FEKETE, B. M. & VÖRÖSMARTY, C. J. 2008. Semantic web infrastructure supporting Next-FrAMES modelling platform. *American Geophysical Union Fall Meeting 2008*, San Francisco.
- LEAVESLEY, G. H., MARKSTROM, S. L., BREWER, M. S. & VIGER, R. J. 1996. The modular modelling system (MMS): the physical process modelling component of a database-centred decision support system for water and power management. *Water, Air and Soil Pollution*, **90**, 303–311.
- LU, B. & PIASECKI, M. 2012. Community modeling systems: classification and relevance to hydrologic modelling. *Journal of Hydroinformatics*, **14**, 840–856, <http://dx.doi.org/10.2166/hydro.2012.060>
- MAKROPOULOS, C., SAFOLEA, E., BAKI, S., DOUKA, E., STAMOU, A. & MIMIKOU, M. 2010. An integrated, multi-modelling approach for the assessment of water quality: lessons from the Pinios River case in Greece. In: SWAYNE, D. A., YANG, W., VOINOV, A. A., RIZZOLI, A. & FILATOVA, T. (eds) *IEMSS 5th International Congress on Environmental Modelling and Software*, 5–8 July 2010, Ottawa, Canada, International Environmental Modelling and Software Society, Manno, Switzerland, 1023–1032.
- MAXWELL, T. & COSTANZA, R. 1997. A language for modular spatio-temporal simulation. *Ecological Modelling*, **103**, 105–114.
- MET OFFICE 2013. The great storm of 1987. <http://www.metoffice.gov.uk/education/teens/case-studies/great-storm> [accessed 4 November 2014]
- MOORE, R. V. & TINDALL, C. I. 2005. An overview of the Open Modelling interface and environment (the OpenMI). *Environmental Science & Policy*, **8**, 279–286.
- NICHOLLS, R. J., BRADBURY, A. ET AL. 2012. iCOASST – integrating coastal sediment systems. In: LYNETT, P. & MCKEE SMITH, J. (eds) *Proceedings of the 33rd International Conference on Coastal Engineering*, 1–6 July 2012, Santander, Spain. International Conference on Coastal Engineering, Los Angeles, 3685–3699.
- NIELSEN, M. 2011. *Reinventing Discovery: The New Era of Networked Science*. Princeton University Press, Princeton, NJ, USA.
- NONAKA, I. 1998. The knowledge-creating company. In: DRUCKER, P. F. (ed.) *Harvard Business Review on Business Management*, Harvard College, Cambridge, MA, USA, 21–45.
- OECD 2004. *Science, Technology and Innovation for the 21st Century. Meeting of the OECD Committee for Scientific and Technological Policy at Ministerial Level, 29–30 January 2004, Final Communiqué*. Organisation for Economic Co-operation and Development, Paris.
- OFFICE OF SCIENCE AND TECHNOLOGY 2004. Foresight Future Flooding. Office of Science and Technology, London.
- OPENMI ASSOCIATION 2010a. OpenMI Standard 2 Specification for the OpenMI (Version 2.0). Part of the OpenMI Document Series. <http://www.openmi.org> [accessed 22 November 2012].
- OPENMI ASSOCIATION 2010b. OpenMI Standard 2 Reference for the OpenMI (Version 2.0). Part of the OpenMI Document Series. <http://www.openmi.org> [accessed 22 November 2012].
- PEARCE, G., MILLARD, K. & HARPER, A. 2010. Application of OpenMI interfacing to promote integrated

- modelling across the water/environment sector, the FluidEarth platform. *Proceedings of the 1st European IAHR Congress*, 4–6 May 2012, Edinburgh, UK.
- PRIGOGINE, I. & STENGERS, I. 1985. *Order out of Chaos: Man's Dialogue with Nature*. Harper Collins, London.
- REED, M., CUDDY, S. M. & RIZOLLI, A. E. 1999. A framework for modelling multiple resource management issues – an open modelling approach. *Environmental Modelling & Software*, **14**, 503–509.
- SAN ROMAN BLANCO, B. L., SUTHERLAND, J. & HARPER, A. 2012. Pyxis: a new workflow manager for integrated modelling. In: HINKELMANN, R., NASERMOADDELI, M. H., LIONG, S. Y., SAVIC, D., FRÖHLE, P. & DAEMRICH, K. F. (eds) *Understanding Changing Climate and Environment and Finding Solutions, Proceedings of 10th International Conference on Hydroinformatics, HIC 2012*, 14–18 July 2012, Hamburg, Germany.
- SAFIOLEA, E., BAKI, S. *ET AL.* 2011. Integrated modelling for river basin management planning, Proceedings of the ICE. *Water Management*, **164**, 405–419.
- SALTELLI, A., RATTO, M. *ET AL.* 2008. *Global Sensitivity Analysis. The Primer*. John Wiley & Sons, Chichester, UK.
- SHRESTHA, N. K., LETA, O. T. *ET AL.* 2012. Integrated modelling of river Zenne using OpenMI. In: HINKELMANN, R., NASERMOADDELI, M. H., LIONG, S. Y., SAVIC, D., FRÖHLE, P. & DAEMRICH, K. F. (eds) *Understanding Changing Climate and Environment and Finding Solutions, Proceedings of 10th International Conference on Hydroinformatics, HIC 2012*, 14–18 July 2012, Hamburg, Germany.
- SUTHERLAND, J., PEET, A. H. & SOULSBY, R. L. 2004. Evaluating the performance of morphodynamic modelling. *Coastal Engineering*, **51**, 917–939.
- TUCKMAN, B. W. 1965. Development sequence in small groups. *Psychological Bulletin*, **63**, 383–399.
- VALKE, S., GUILVARDI, E. & LARSSON, C. 2006. PRISM and EMES: a European approach to earth system modelling. *Concurrency and Computation – Practice and Experience*, **18**, 247–262.
- VAN DER WAL, T. & VAN ELSWIJK, M. J. B. 2000. Generic framework for hydro-environmental modelling. *Proceedings of the 4th International Conference on Hydroinformatics*, 23–27 July 2000, Cedar Rapids, IA, USA. IWA Publishing, London.
- VOINOV, A. & CERCO, C. 2010. Model integration and the role of data. *Environmental Modelling & Software*, **25**, 965–969.
- VON BERTALANFFY, L. 1951. General systems theory – a new approach to unity of science. *Human Biology*, **23**, 303–361.
- WHELAN, G. & NICHOLSON, T. J. (ed.) 2002. Proceedings of the Environmental Software Systems Compatibility and Linkage Workshop. NUREG/CP-0177, PNNL-13654. US Nuclear Regulatory Commission, Rockville, MD, USA. <http://www.nrc.gov/reading-rm/doc-collections/nuregs/conference/cp0177/> [accessed 31 May 2013].
- WHITEHOUSE, R., BALSON, P. *ET AL.* 2009. *Characterisation and prediction of large scale long-term change of coastal geomorphological behaviours: final science report*. Science Report SC060074/SR1. Environment Agency, Bristol, UK.