International Congress on Environmental Modelling and Software

4th International Congress on Environmental Modelling and Software - Barcelona, Catalonia, Spain - July 2008

Jul 1st, 12:00 AM

Delft FEWS: A proven infrastructure to bring data, sensors and models together

Peter Gijsbers

M. G. F. Werner

J. Schellekens

Gijsbers, Peter; Werner, M. G. F.; and Schellekens, J., "Delft FEWS: A proven infrastructure to bring data, sensors and models together" (2008). *International Congress on Environmental Modelling and Software*. 89.
https://scholarsarchive.byu.edu/iemssconference/2008/all/89

iEMSs 2008: International Congress on Environmental Modelling and Software
Integrating Sciences and Information Technology for Environmental Assessment and Decision Making
4th Biennial Meeting of iEMSs, http://www.iemss.org/iemss2008/index.php?n=Main.Proceedings
M. Sànchez-Marrè, J. Béjar, J. Comas, A. Rizzoli and G. Guariso (Eds.)
International Environmental Modelling and Software Society (iEMSs), 2008

Delft FEWS: A proven infrastructure to bring data, sensors and models together

P.J.A. Gijsbers[a] , **M.G.F. Werner**[a,b] **and J.Schellekens**[a]

[a] *Deltares-Delft Hydraulics, Rotterdamseweg 185, 2629 HD, Delft, The Netherlands*
(peter.gijsbers@deltares.nl)

[b]*Department of Water Science & Engineering, UNESCO-IHE, P.O. Box 3015, 2601 DA Delft,*
the Netherlands

Abstract: Delft FEWS is a proven real time software infrastructure for operational water management and forecasting, having been applied in several operational forecasting centres all over the world. The system has contributed to changing the paradigm of flood forecasting systems from a model centric approach to a data centric approach. The system is a time series oriented ETL-infrastructure (Extract, Transfer, Load) with the ultimate aim to provide the forecaster with relevant information in support of the operational process of flood forecasting and warning. Key to its success is the design choice for separation of data communication protocols from the content, as well as its openness to integration of other applications and data. While keeping its focus on its core tasks, the design concept has allowed the system to evolve from a stand alone application, through a client-server concept into a (web) service oriented architecture. Continued research is ongoing to explore new technologies and turn these into operational features available to operational forecasters. This paper will highlight some of our experiences in system design to combine real time data with models.

Keywords: FEWS; ETL; real time; forecasting; modelling; time series; data; sensors

1. INTRODUCTION

Delft FEWS is a proven, state-of-the-art, real time software infrastructure for operational water management and forecasting. The system is a sophisticated collection of modules designed for building an operational water management system customised to the specific requirements of an individual agency. The philosophy of the system is to provide an open shell system for managing the operational management process [Werner et al.2004]. This shell incorporates a comprehensive set of general data handling utilities, while providing an open interface to a wide range of forecasting models. The modular and highly configurable nature of the system allows it to be used effectively both in rudimentary systems and in highly complex systems utilising several simulation models. Delft FEWS can either be deployed in a stand-alone, manually driven environment, as a fully automated distributed client-server application, or as a web-service component within another forecasting system.

Over the last few years, Delft FEWS has been implemented as the prime operational system in several operational flow forecasting systems used both by national authorities in for example England, Wales, Scotland, Germany, Switzerland, Taiwan and the Netherlands, and by regional authorities in amongst others the Netherlands, Austria, Thailand and Italy. In addition to these, several operational and pilot systems are currently being implemented in the Netherlands, USA, Singapore, Russia, Spain and in the Mekong basin. Its expansion has, however, not been constrained to a geographic expansion as more recently its range of application has evolved from a focus on the use in operational flood forecasting systems to a much wider application range for operational water management forecasting of both high and low flows, reservoir and storm surge barrier operation, groundwater management, algae bloom and water quality forecasting. In addition it is also being applied as a Water Information System, where it forms the heart of an infrastructure that focuses on automated data collection, validation, visualization, and persistent storage.

This paper will discuss the philosophy behind Delft FEWS. It will analyse the design choices that have enabled the expansion and it will address some of the lessons learned in implementation across such a wide range of domains.

2. THE PHILOSOPHY BEHIND Delft FEWS

2.1 Support the operational team

Flood Early Warning Systems, from which the FEWS acronym has been derived, have been established for many years [see e.g. Grijssen et al, 1992]. These have a primary role in providing comprehensive information to the operational forecaster in a concise and efficient manner. It is then the responsibility of the operational forecasting team to take guidance from the information provided, and when relevant issue a warning to relevant authorities and if applicable the general public. From the perspective of this role in support of the operational flood forecasting, warning and response process [Parker & Fordham, 1996], describing the system as a warning system is somewhat misleading, as the warning itself is issued by the forecasting team, having taken guidance from the real time data and the forecasting system. Within this perspective, the main purpose of a forecasting system in support of the warning process, as compared to when a monitoring system only is used, is to increase the lead time of warnings so that authorities can act upon expected conditions [see Werner *et.al*, 2005], Werner & Heynert [2006] discuss this role in more detail. Within this perspective, two key factors should be mentioned that have had significant impact on system design: (i) robustness and (ii) confidence.

2.2 Deliver robustness and confidence

As discussed, the primary role of a forecasting system in the operational setting is to provide data to be used as guidance to the forecaster in making a decision on whether or not to take action. Within this context, the confidence that the forecaster vests on the data provided is essential if that information is to be used meaningfully. Typically the confidence of operational forecasts are addressed in terms of how trustworthy observations and forecasts are, how large uncertainties in the forecasts made is and, in that same light, how skilful these forecasts are in predicting forthcoming events.

These considerations on factors that contribute to forecaster confidence are obviously important, but additional factors such as the robustness of the system itself will also need to be considered. Robustness refers to the fact that the system should offer duty forecasters the latest and best data available, given the conditions in the field. If the system fails, no or very little information may be available, and decisions may need to be taken unsupported by data. Should the system fail regularly, forecasters will naturally lose confidence in its ability to provide relevant information, and seek guidance from other sources. Much attention has been paid to this aspect in the design of Delft FEWS. Robustness has been one of the prime drivers in establishing a highly modular concept. As there is a high degree of independence of these modules, some may be allowed to fail and lead to degradation in the information provided. The modularity ensures this will not cause a complete failure of the system.

2.3 Focus on data handling and provision, not on running models

Many flood forecasting systems used to be, and sometimes still are, based on a model centric approach [Werner & Whitfield, 2007]. Having been established essentially as a wrapper around a specific model or models, the execution of the hydrological and hydraulic models seemed to be the main task of these systems and as such every item was designed for this task. Operational users, however, are not always focused on the models themselves but rather on gaining insight in a variety of data streams, ranging from tidal surge data and in-situ observations through to meteorological and hydrological forecasts. A key task for an operational information system is to bring all information together and create a consistent and reliable information base for decision making. The collection, validation, transformation, storage and visualization of data play an equal role next to the running of models. It should be stressed that efficiency in combining data from several sources in differing ways efficiently is paramount to allow the forecaster gain a rapid understanding on the basis of the data available.

2.4 Focus on essential system tasks, not on domain specific content or formats

Tasks such as data collection, validation, transformation, storage and visualization are common to many domains in and outside the flood forecasting or even the wider water domain. System-infrastructures that conduct these types of tasks are typically called ETL-infrastructures, i.e. extract data from external sources, transform this into a consistent format that can be loaded to other applications or components.

A characteristic of hydrological forecasting systems is the spatial and temporal nature of the data that can be handled. Delft FEWS has no additional domain specific knowledge to conduct all data processing tasks. Its design choice to strictly separate the water domain from the system design and to separate data/communication protocol and content is considered a major contributor to its evolution from its application to the flood forecasting domain towards its current wide range of application.

2.5 Create openness for other system components

Water authorities often have knowledge bases and computer models that have been developed over time. When adopting a new comprehensive forecasting system they will be reluctant to throw these components away, since much knowledge and experience is vested in these models and knowledge bases, which may also represent large financial investments. Incorporating such existing knowledge components in a new system that in essence provides only an operational framework is a considerable asset. Not only does reuse of existing components ensure that both knowledge and financial investments are maintained, their reuse also bolsters acceptance of and confidence in any new system. Openness in system design is therefore considered essential. It is also an asset that reduces supplier dependency for new system components (e.g. models). This need for open systems is discussed in detail by Werner & Heynert [2006], and its value is also reflected in e.g. Funke et al. [2006]. The separation of form and function also makes it easy to identify interfaces that create openness to existing system components or new system components from other suppliers.

2.6 Deliver product stability

Operational forecasting teams typically encompass a large number of qualified staff to be able to provide a continuous service. As a consequence, the stability of the system in terms of how staff interacts with it are of major concern, particularly for those systems that are used nationally in several operational centres. The investment associated with developing forecasting and warning procedures, documenting these procedures and training relevant staff is extensive for organisations such as the Environment Agency of England and Wales [Whitfield, 2005]. Werner & Heynert [2006] have highlighted how an open system enables improvement of the forecasting and data processing capabilities with minimal impact on how operational staff uses the system on a day to day basis. As a consequence, product developers should ensure that the appearance of the system to its direct users, in particular the user interface, does not evolve too fast, as agencies will be reluctant to repeatedly send large numbers of staff on training to understand such ‘improvements’. In the same line of thought, backward compatibility should be ensured and plans carefully made for removing the utilization of functionality that has been superseded by other components.

2.7 Hand back responsibility for system extension

Many agencies are all too aware of dependency on external suppliers. They often are anxious for vendor-locks, i.e. contracts and products where they are bound to one supplier for a long period over time. The ability to separate the modelling task from system configuration enables agencies to outsource the modelling work to other suppliers. Once trained in system configuration, they also can take responsibility for that configuration. In this way, they are in control again, with only minor dependency on the supplier for basic system maintenance and support. Deltares-Delft Hydraulics, the system supplier of Delft FEWS, is keen to stimulate agencies to take responsibility for their Delft FEWS application as innovation in forecasting techniques will then not only be driven by the ideas of a central supplier only, but will be complemented by all users who are actively involved in configuring and working with the system. The aim of Deltares-Delft Hydraulics is then to act as an intermediary in sharing innovation across users.

3. SYSTEM FUNCTIONALITY

In functional terms, data processing within Delft FEWS is guided by the ETL-paradigm; extract data, transform into a consistent format and load into forecasting models. These functionalities are provided by four modules that work on top of a high performance data access layer which combines relational database technology with a highly compressed data structure that can handle large sets of time series data, including spatial time series.

3.1 Extract data

The import module provides the database with a quality controlled data set, obtained from external sources, such as telemetry systems, remote sensing observations (radar and satellite) and (ensemble) weather predictions. The design of this import module has focussed on facilitating automated detection of missing values and validation against prescribed ranges and maximum rates of change.

There are currently no standards in data formats made available through telemetry systems, and as a consequence a range of data formats need to be catered for. In Delft FEWS a design pattern is chosen that requires a (small) code extension for each new format to be considered. This can be done within the Delft FEWS code base but also by registering a Java class that has been made according to the Delft FEWS import plug-in specifications. The latter option does not require the original code developer to be involved. This approach was chosen in preference to the development of a generic import class that could be configured to the specifics of a format. Experience with such generic classes has shown that these quickly become overly complex as there is in principle an unlimited combination of options. Establishing a targeted class specific to a format has been found much more efficient and resilient. For remote sensing data and weather predictions industry standards such as GRIB and BIL/BSQ are, however, supported.

The default operation method of this import module is to fetch data at regular intervals from a disk device within the LAN. Recently, the module has been extended to acquire data from an external server (FTP or HTTP). However, the preferred approach is often to keep the import mechanism used as simple as possible, with the simple file based mechanism often being the most appropriate. More complex modes of operation, e.g. direct database queries or web service requests are available, but often not considered appropriate as for example low server response may result in time-outs or blocking of other server processes. Essentially experience has shown that tight integration of separate systems using complex data transfer mechanisms may be technically desirable, but may be detrimental to overall robustness due to dependency on multiple systems being fully available simultaneously. Within the sensor world, initiatives are being undertaken to develop standard data exchange formats for sensor data. The Sensor Web Enablement protocol (SWE), developed by the Open Geospatial Consortium [OGC, 2006] is one of these initiatives. This protocol has been evaluated in a research project co-funded by the Dutch program Ruimte voor Geo informatie. After a test implementation to query various Sensor Observation Services for observation data, we concluded that this protocol has various drawbacks when used within a high-performance operational setting as is required for forecasting purposes [Jellema, 2008]. The most important drawbacks found where:

- The reliance for SWE on GML (Geographic Markup Language) creates a large overhead in meta-data, which results in large XML-objects to be passed, while the actual data of interest, i.e. the observations, are relatively small in package size.
- The key data content within the XML-string, i.e. the observed values, is packed in a generic XML-element which has to be interpreted based on a self-described format. A dedicated, and complex, piece of code is required to parse this element in a generic way, since it cannot be handled by an industry standard XML-parser.

3.2 Transform data

Typically, the data imported needs to be fed into forecasting models. Spatial and temporal resolutions need to be adapted and gaps need to be filled to ensure that the data fits the input requirements of the model. Within Delft FEWS the functionality required for manipulation, combination and completion of data are implemented by an interpolation module and a transformation module. The interpolation module offers a wide range of spatial and temporal interpolation algorithms, while the transformation module offers a wide range of functions to transform (complete) series. The transformation module can be considered as the work-horse of the system, providing an extensive library of data transformation options. Although some of these have a hydraulic or hydrological background, most are domain independent.

3.3 Load into models using an open interface format

Once prepared, the data can be loaded into the forecast model to run. Delft FEWS uses an open interface approach to accommodate models from various suppliers. A standardized XML-format, the so-called Published Interface, is used to exchange the relevant data with the models. From the system's perspective, the model run is managed by the so-called General Adapter (see Figure 1). For each model run, the General Adapter exports the relevant model state (in native format) and boundary conditions (in XML) to the model. A tailored Model Adapter converts this data into the native model format before the model is kicked off. After execution has completed, the Module Adapter converts the results back into the Published Interface-format. Finally, the General Adapter imports these results as well as the update (native) model state file, which is stored for reuse at the next run.

By adopting this standardized and open approach, the General Adapter only needs to know the data exchange format while all model specific knowledge can be encapsulated in the model adapter. The use of the published interface standard allows such model adapters to be established wholly independent of the Delft FEWS system itself, thus ensuring independence of the suppliers of the model code from the suppliers of the Delft FEWS system, should such independence be desirable from a contractual perspective.

Model codes from a range of model developers have been integrated with the approach described. Werner & Heynert [2006] published a list of some 19 models that were available at the time, but this list has been extended to include several other codes, such as HEC-ResSim, HEC-RAS, Hec-HMS, Modflow, Vflo, TOPKAPI, SNOW17, SAC-SMA, Ribasim, Delft3D-Flow, SOBEK-WQ and Delft3D-WQ.

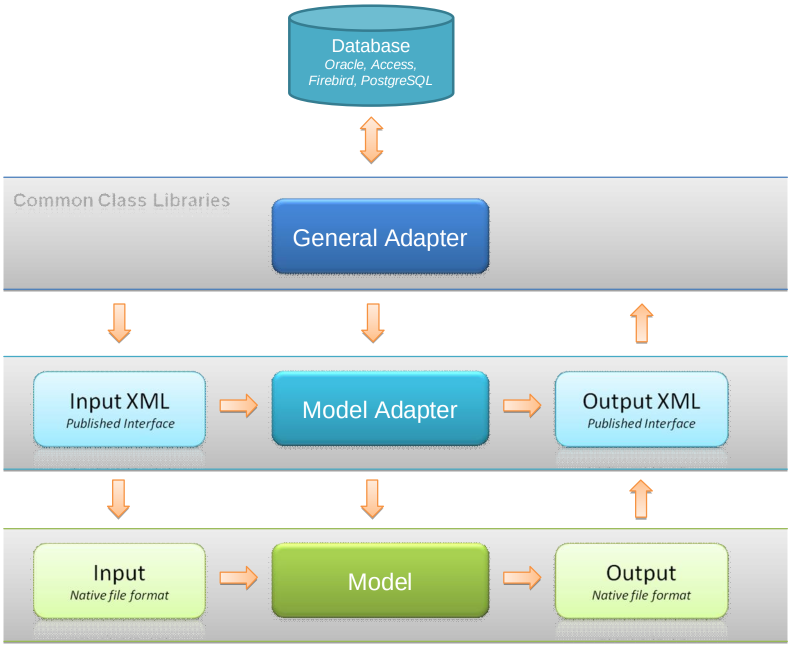

Figure 1 Open communication between Delft FEWS and Models

4. SYSTEM IMPLEMENTATION

Delft FEWS is a scalable system developed with J2EE technology. Stand alone applications utilize the Delft FEWS-shell instance as a combined GUI and computational server. This instance has its own local database. In the client-server setting, these two roles are separated. The same Delft FEWS-shell instance is then utilized on the client side as a thick client application, referred to as the Operator Client (OC) through which the data is disseminated to the forecaster, and on the served side, referred as a Forecasting Shell Server (FSS). Forecasting Shell Servers act as the computational engine which conducts all data processing tasks from importing data to running models and, if desired, producing HTML- reports. Data produced by FSSs is synchronized to a central database. From this database, data can then be distributed to the various client-applications. At the heart the system is the Master Controller (MC). This component manages the communication between all components using JMS. It contains a job scheduler to automatically initiate tasks on the FSSs. If a multiple-site system is deployed, the Master Controller ensures data replication at all sites either in duty standby mode or in replication mode to balance the load. A dedicated web-interface is available to manage the Master Controller and maintain the system. To manage the system configuration, a separate maintenance component is available, while all system components will detect configuration updates automatically. Recently, a fully automated archiving sever has been added as well as a web server to integrate forecast reports from various forecasting agencies. The product openness has been extended with a JDBC driver to directly obtain time series or associated graphic figures. Alternatively, a web service may be requested to obtain data via the Delft FEWS Published Interface formats. The web-service can also be used to kick off a task to the Master Controller.

5. ROBUSTNESS AND CONFIDENCE

The utility of an operational system would be considered low by any operational team if the system fails when it is required most, during for example a flood event. Repeated failure of the system at such times will quickly erode any confidence in it. Such confidence can be built up by ensuring that the system runs properly with a very high availability. For example, the choice for a fat-client with a local database over a thin (web) client is guided by a desire to reduce dependency on continuos availability of high-performance internet connections. During implementation, various precautionary actions can be taken to increase system robustness, such as: ensure a continuous power supply, independent of general power failures, prevent flooding of the forecasting centre, provide back up or dual site implementation, apply proven technology, and ensure proper support arrangements. Confidence is also increased if the data provided is available, useful, frequently updated and properly flagged in terms of reliability. The latter requires thorough understanding of the forecasting process to ensure that the system does not stop should data not be available to the full extent normally expected. Fall back options to use other data sources, should be available if required.

5.1 Fall back options

Numerical simulation models are normally very strict in their input requirements. They typically cannot handle missing data in their input data sets. If data import, validation and transformation fail to provide the required input for simulation models, alternative input sources may be chosen or the model run might be skipped. In the latter case, rudimentary lookup tables or correlation functions might become the base to deliver an answer to the operational team using the best available knowledge combined with available data sources. This so-called ‘data hierarchy’ concept is crucial for system robustness as it allows alternative data sources to be used as a fallback in the forecasting process should the nominal sources be unavailable. The use of these options will ensure the continuity of the forecasting process, even if available real-time data is incomplete or inconsistent. As such robustness contributes to confidence as the operational team is aware that the system will always provide the best and latest data available, including associated quality flags.

5.2 Confidence in forecasting models

Skilful forecasts play a vital role in building confidence. Flood events often happen with model conditions that often are not very well tested due to limited data availability. Various advanced techniques, such as error correction and ensemble Kalman filtering, can be applied to improve model performance.

Methods to evaluate model performance form a key component in evaluating the performance of the forecast models. Within Delft FEWS, the concept of establishing performance indicators as an integral part of the forecast process has been introduced to compare observed and calculated time series of the same type at a given location. This accuracy can be assessed at the level of the individual forecasting modules using performance measures typically applied in module calibration (e.g. root mean square error,

Nash-Sutcliffe). In addition, the performance of the forecasting system itself can be derived to reflect the accuracy of the system in forecasting. Three types of measure are available; (i) lead time accuracy of forecast time series, (ii) accuracy of timing threshold event crossings, and (iii) accuracy and timing of peak predictions [Delft Hydraulics, 2005]. Confidence of operational teams also improves if both the false alarm rate and the number of missed events are low. For this purpose a forecasting skills concept is used. Indicators such as probability of detection and false alarm ratio are derived from a contingency table in which the number of (calculated) threshold crossings is compared against the number of observed threshold crossings.
Within Delft FEWS, both the performance indicators and the skills score table are continuously updated as a part of the operational process. Integration of such evaluation measures can be used both to target improvements where forecasting skill is considered low, as well as to increase model complexity only when proven to increase forecast skill. This allows for an optimal balance between CPU run time and the spatial model representation. Models with high spatial resolution require more computation time, but may not necessarily lead to better results for forecasting purposes. Often, it is preferable if the frequency of runs is determined by the availability of data and not by the runtime of the model.

6. LESSONS LEARNED

Operational systems for operational water management have to support the operational team with reliable data provision, both real time monitored and forecasts. Skilful forecasts and a system design accounting for robustness have a significant impact on confidence in the system. Continuous assessment of forecast skills and performance can contribute to decisions where investments in the system are most effective. Openness in system design allows agencies to reuse previously developed components, e.g. models and knowledge bases, hence enabling preservation of previous investments. The design choice to separate domain knowledge from the system modules and to separate data/communication formats from functional content is considered a major contributor to the evolution of Delft FEWS from its application in flood forecasting towards its current wide range of applications.

REFERENCES

Delft Hydraulics, DelftFEWS Configuration Guide, 2005. Also available at http://public.wldelft.nl/display/FEWSDOC/Home

Funke, R., B.Courtois and S.Soleilhavoup, Integrated hydrological information and knowledge management transfer for flood warning centres in European countries. Proc. of the 7th Int. Conf. on Hydroinformatics, 2006. Gourbesville, Cunge, Guinot and Liong (eds), Research Publishing, Vol.1, pp.163-170

Grijsen, J., Snoeker, X., Vermeulen, C., el Amin Moh. Nur, M., and Mohamed, Y., An information system for flood early warning, In: A. Saul (ed.), Floods and Flood Management, Kluwer Academic Publishers, (1992) 263–289.

Jellema, J. SWE-connected sensors: data source for geo information. European Spatial Data Research & Int. Soc. for Photogrammetry and Remote Sensing workshop on Geosensor Networks. Hannover, Germany 2008, http://www.ikg.uni-hannover.de/geosensor

OCG OpenGIS® Sensor Observation Service Implementation Specification, OGC® 06-009r1, OpenGIS Implementatoin Specification Open Geospatial Consortium Inc. 2006

Parker, D. and Fordham, M., Evaluation of flood forecasting, warning and response systems in the European Union. Water Resour. Manage. **10**, (1996), 279–302

Werner, M., M. van Dijk and J.Schellekens Delft-FEWS: an open shell flood forecasting system; 5th International Conf on Hydroinformatics. Liong, Phoon & Babovic (eds) 2004 World Scientific Publishing Company Vol.2 pp.1205-1212.

M.G. Werner J. Schellekens and J.C. Kwadijk Flood early warning systems of hydrological (sub) catchments. Encyclopedia of Hydrological Sciences, 2005. M.G. Anderson and J.J. McDonnell (eds), 349- 364

Werner, M.G.F. and K. Heynert, Open Model integration – a review of practical examples in operational flood forecasting, Proc. of the 7th Int. Conf. on Hydroinformatics, 2006. Gourbesville, Cunge, Guinot and Liong (eds), Research Publishing, Vol.1, pp.155-162

Werner, M., and D. Whitfield, On model integration in operational flood forecasting, Hydrol. Process. **21**, (2007) 1519–1521.

Whitfield D., The National Flood Forecasting System (NFFS) of the UK Environment Agency. In: Int. Conf. on innovation, advances and implementation of flood forecasting technology. Balabanis P, Lumbroso D, Samuels P (Eds). Tromsø, Norway, (2005), http://www.actif-ec.net (last accessed 7-8-2008).