Incorporate Sensor Data and Dynamic Modeling for Real Time Control of Sewer Systems

Erik Rooij, Klaas-Jan Heeringen

Incorporate Sensor Data and Dynamic Modeling for Real Time Control of Sewer Systems

Erik de Rooij and Klaas-Jan van Heeringen

Deltares, P.O. Box 177, 2600 MH Delft, The Netherlands
{erik.derooij, klaasjan.vanheeringen}@deltares.nl

Abstract. We discuss how Delft-FEWS is applied to incorporate real-time sensor data and dynamic modeling to allow Real Time Control (RTC) of three sewer systems in the area of the Hoeksche Waard, the Netherlands. The purpose of this project was to reduce sewerage spilling to the surface water by optimizing the available storage in the sewer systems. In this paper we will present how Delft-FEWS makes use of Simple Object Access Protocol (SOAP) webservices to exchange real-time data with multiple sewer systems and their different Supervisory control and data acquisition (SCADA) systems. We will also explain how we combine this with dynamic models, allowing for central automatic control of multiple sewer systems.

Keywords: Delft-FEWS; Real Time Control; Webservices; SOAP; SCADA systems; dynamic modeling; Sewer Systems; Sensor Web

1 Introduction

Over the last decade Delft-FEWS has been applied in a wide range of operational water management and forecasting systems. In this case study Delft-FEWS is used for the first time to integrate dynamic models with automatic system control. This introduces new challenges concerning; real-time availability of sensor data, fail safe mechanisms in case of network interruptions and fast response times to allow for automatic control. In this paper we discuss how we tackled these challenges, experimenting with different mechanisms for exchanging sensor data between the SCADA systems that are operated at the municipalities and the Waste Water Treatment Plants (WWTP) and Delft-FEWS.

1.1 Description Delft-FEWS

Delft-FEWS is a proven, 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 customized to the specific requirements of an organization. The philosophy of the system is to provide an open shell system for managing the operational management process [1-2]. 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 configurable nature of the system allows it to be used both in rudimentary systems and in highly

adfa, p. 1, 2011.

complex systems. Delft-FEWS can be deployed as stand-alone environment, as a fully automated client-server application, or as a web-service component within another forecasting system.

1.2 Case Study Hoeksche Waard

For this case study the regional water authority Hollandse Delta and five municipalities (Binnenmaas, Cromstrijen, Korendijk, Oud-Beijerland and Strijen) (see Fig. 1) are exploring ways to increase the efficiency of the waste water treatment process and to improve water quality [4]. Together with Deltares and Delft University of Technology they started three pilot projects aimed to centralize automatic control of the pumps in the sewer systems of the municipalities and the pumps that transfer sewage water to WWTP.

Each of the municipalities and the WWTP operate their own SCADA system. These systems collect the data from the sensors within the sewer system and display this to the operators. From these displays the operators can monitor and control their part of the system. The SCADA systems of the different municipalities and WWTPs are not connected. By introducing Delft-FEWS as a central control component we hope to connect the SCADA systems and achieve the following objectives.

The main objective of this project is to reduce combined sewerage overflow (CSO) volumes from the sewer systems to the surface water. This objective is achieved by optimizing the remaining available storage within the sewer system during storm events. Another objective is the reduction of pumping costs and waste water treatment costs [3].

Fig. 1. Hoeksche Waard area with the five municipalities

2 Centralizing Control by Creating a Sensor Web

The Netherlands have an extensive sewer network that collects waste water from households, industries and urban overland flow. To transport the waste water through the system towards the WWTPs an extensive network of sewer pumps is maintained. The capacity of the sewer system is limited and during storm events the system can fill up, resulting in harmful sewage spills to surface waters.

Each municipality is responsible for maintaining and operating the pumps within its boundaries, while the regional water authority is responsible for operating the WWTPs. In general the pumps work independently and do not take into account the levels within the rest of the system. Once the sewage exceeds the local start level the pump will commence operation and will continue until the level drops below the local stop level. Each sewer pump is linked to a SCADA system from which it can be monitored and controlled. The SCADA systems of the municipalities and WWTPs are not linked together.

By linking the SCADA systems of the municipalities and WWTPs together and by introducing a dynamic modeling component it becomes possible to optimize the sewage flow throughout the whole system and at the same time allows for an automated centralized control of the system.

To accomplish this webservices are used to link the SCADA systems together and Delft-FEWS is introduced as dynamic modeling component to be the central information system. Data is collected from each SCADA system and is fed to Delft-FEWS, where a RTC plugin calculates the optimal pump settings. These settings calculated from Delft-FEWS are sent back to the SCADA systems in order to control the sewer pumps.

2.1 Automatic Control Requirements

The introduction of the 'automatic control' aspect results in extra requirements to ensure the safety of the sewer system and pumps:

- *Fail over mechanism*: In the case of network interruptions, or on receiving incorrect pump settings the local system must react to this by switching from automated control back to local control to prevent damage to pumps or cause overflowing to surface waters.
- *Secured against external interference*: The system must be protected against external malevolent parties who try to 'hack' into the system and take control of the pump stations. Firstly this should be prevented by using adequate security measures such as: user authentication, limit access to known IP-addresses and communication over HTTPS. Secondly the fail over mechanism should kick in as soon as harmful control settings are passed to a pumping station.
- *Real-Time data availability*: In order for the dynamic models to produce any meaningful control output it is important that the input data is as close to real-time as possible. Also the return period of the calculated control parameters must be well within the system response time. If either of these requirements is not met the sys-

tem will fail over. Experience shows that a response time equal to or less than 30 seconds is required.

3 Creating a Sensor Web with Webservices

As described earlier in this paper, the SCADA systems of the municipalities and WWTPs reside on individual networks not connected in any way. Each municipality is responsible for maintaining and operating its own SCADA system. None of the municipalities are very eager to open up their system to outsiders and relinquish control. In order to create a Sensor Web from these individual and closed systems it is necessary to minimize the required changes and risks to the local systems. By using webservices the impact to the local systems is kept to a minimum:

- Open a communication port over which all HTTP or HTTPS traffic is sent,
- Install a software component locally on each individual SCADA system that enables the SCADA system to interact with the Sensor Web.

During the course of the project two mechanisms are implemented for transferring data between the SCADA systems and the central controller Delft-FEWS (see Fig. 2):

1. SCADA data is sent to a webservices hosted by Delft-FEWS (FewsPiService).
2. Delft-FEWS interacts with a webservice hosted on the side of the SCADA system.

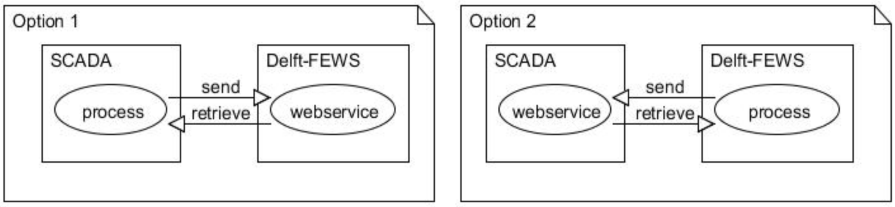

Fig. 2. Overview transfer mechanisms

Both of these mechanisms have to be able to interact with the following two SCADA system types:

- Control Maestro: Produced by Elusions SAS (www.elusions.com) and uses OLE for Process Control (OPC) technology.
- Flygt (RTiem): Produced by Xylem Nederland (www.xylemwatersolutions.com).

3.1 Webservice Hosted by Delft-FEWS

At the start of the project a decision is made to make use of the webservice provided by Delft-FEWS, the FewsPiService1, which exchanges data using the Published In-

[1] http://publicwiki.deltares.nl/display/FEWSDOC/Fews+PI+service

terface (PI) XML format2. On the SCADA system side this requires the installation of a JAVA process that can exchange data between the SCADA system and the Delft-FEWS webservice. This process is called the Data Transfer Service (DTS) and requires a local configuration file containing information about the data to transfer and at what frequency the data should be transferred. For example one type of data can be scheduled every second while the other is scheduled once a day, at a certain time. Besides the sending of SCADA data, the DTS also sends log messages produced during data retrieval. Furthermore during network interruptions the DTS stores the data locally in order to re-send it once the network connection is re-established. A schematic representation of this setup can be seen in Fig. 3.

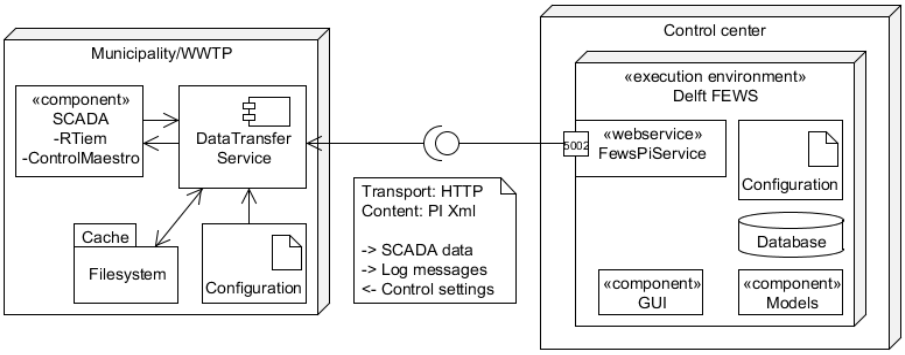

Fig. 3. Schematic representation of webservice hosted by Delft-FEWS

Advantages of this approach:

- At the time of implementation, this approach had the advantage of smaller response times, due to the Delft-FEWS system not being able to cope well with model runtimes less than one minute.
- Because data is sent at regular intervals it is possible to build a continuous historical archive of the SCADA data. Even during network interruptions the data is cached and is sent later on. This is convenient hence SCADA systems do not always store their historical data.
- By being able to tune the transfer frequencies it is possible to reduce network load and the amount of queries on the SCADA system.
- It is easy to manage and send log messages as each process has a dedicated destination.

Disadvantages of this approach:

- Because the DTS process does not run synchronously with the running of the dynamic models, data often arrives slightly too late for the model run or is retrieved

[2] http://publicwiki.deltares.nl/display/FEWSDOC/The+Delft-Fews+Published+interface+%28PI%29

slightly before completion of the model run. Resulting in the output control settings lagging in time.
- Because the configuration of the DTS is located on the local SCADA systems, updates always require the assistance of the system operator of the municipalities. This always results in delays.
- It is not possible to transfer data using HTTPS, as the Delft-FEWS webservice does not support this.

3.2 Webservice Hosted on SCADA System

Later on in the project a decision is made to transfer the control of data exchange to Delft-FEWS. This became possible due to improvements in performance of Delft-FEWS allowing it to run the models as frequent as two to three times a minute. The main reasons for wanting to transfer control are:

- Synchronizing data exchange with model runs.
- Transferring control of configuration to Delft-FEWS

For this change a PI webservice is created that is hosted in a Tomcat web container installed on the SCADA system. Now each time Delft-FEWS performs a model run data is first retrieved from the webservice on the SCADA system, then the dynamic model is run and finally the results are sent back to the SCADA system. For each SCADA system an implementation of the new PI webservice is required. Delft-FEWS does not have any dependencies with the SCADA systems as it communicates through the interface hosted by the PI webservice. Now all configurations concerning data transfer and scheduling is controlled by Delft-FEWS. A schematic representation of this setup can be seen in Fig. 4.

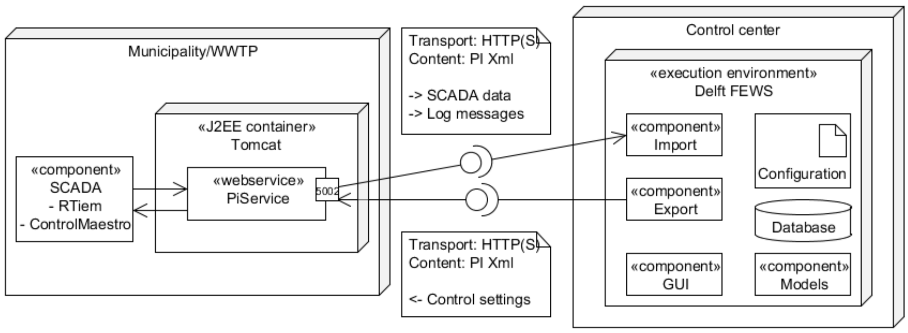

Fig. 4. Schematic representation of webservice hosted on SCADA system

Advantages of this approach:

- Model run always uses the latest data and returns data as soon as it becomes available.
- Configuration resides within Delft-FEWS. It is even possible to update the webservice component through the manager console of Tomcat.

- It is possible to communicate over HTTPS using the security configuration available within Tomcat.
- It is still possible to tune the transfer frequencies by scheduling the import process at different rates.

Disadvantages of this approach:

- There is no longer a caching mechanism. So whenever the network connection is interrupted a gap will appear in the historical archive. If the SCADA system allows historical queries of the data the gap can be filled later during a manual import run.
- Because it is more flexible to query the SCADA system for historical data (if supported by SCADA), it introduces the risk of OutOfMemory exceptions when querying a too large period. At present no mechanism has been built into the system that splits the results into smaller segments. The only security against this is to report the requested data volume is too large and to return only a partial result set.
- It is harder to manage log messages because logged information has to be stored until client retrieves this explicitly in a separate call. If not managed this could lead to memory issues.

4 Project Results

Prior to this project, the Hoeksche Waard sewer system consisted of numerous individual sensor networks. By linking these networks a greater insight was gained in the actual working of the network as a whole. This revealed that the designed specifications of the sewer system did not correspond to the actual situation.

The improved monitoring resulted in a better quantification of the CSO. According to the improved model results there was 48% more CSO than estimated. However because of these new findings it was also possible to implement some simple and cheap improvements.

The goal to reduce pumping costs was not achieved. As a result of the main goal, to reduce emissions, it was not possible to operate the sewer pumps in a more energy efficient way. The only possible solution for reducing pumping costs would be to reduce the inflow of rainwater into the sewer system.

Furthermore flow measurements showed that about two times more water than estimated is entering the sewer system and is being processed by the WWTP's. This leaves a lot of room for improvements and cost reductions.

5 Conclusions and Future Challenges

This project demonstrates how relatively small adjustments to existing SCADA systems allows them to be converted into centrally controlled operating systems. Currently this project is running as an operational system and plans exist to expand the functionality to other municipalities.

During this project we found that when using a Sensor Web there are inherently going to be network issues or partial system failures. It is therefore important to have a robust backup mechanism in place to assure the integrity of the vital system components.

When transferring SCADA data, package size is usually not a concern, as the amount of data transferred is small. But when it is possible to query larger amounts of data, memory management will become a concern. For the future it would be good to investigate mechanisms that support the transfer of large volumes of data, either by segmenting the results or by streaming them.

Acknowledgments. This project was supported by grants from Agentschap NL, The Netherlands. This agency provided a subsidy from the "Innovatieprogramma Kaderrichtlijn Water" of the Dutch Ministry of Transport, Public Works and Water Management. This work is part of the project "Integrale Sturing Afvalwaterketen en watersysteem Hoeksche Waard".

References

1. Werner, M.G.F., Dijk, M. van, Schellekens, J.: Delft-FEWS: An open shell flood forecasting system. In: 6th International Conference on Hydroinformatics, pp. 1205-1212. World Scientific Publishing Company, Singapore (2004)
2. Gijsbers, P.J.A., Werner, M.G.F., Schellekens, J., Delft-FEWS: A proven infrastructure to bring data, sensors and models together 650. In: Proceedings of the iEMSs Fourth Biennial Meeting: International Congress on Environmental Modelling and Software, Volume 1, pp. 28-36. Barcelona, Catalonia (2008)
3. Heeringen, K.J. van, Loenen, A. J. van, Nooyen, R. van, Velsen, E. van: Leveraging existing infrastructure for central automatic control of multiple sewer systems. In: 10th International Conference on Hydroinformatics. Hamburg, Germany (2012)
4. Nooijen, R. R. P. van, Kolechkina, A.G., Heeringen K-J. van, Velzen, E van: Implementation of central control for multiple sewer systems. In: 12th International Conference on Urban drainage, pp. 10-15. Porto Alegre, Brazil (2011)