A Flexible Approach to Information Sharing in Water Industries

Hamideh Afsarmanesh, Ammar Benabdelkader, and L.O. Hertzberger

University of Amsterdam, WINS - Department of Computer Science,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
Phone: +31 20 525 75 12, Fax: +31 20 525 74 90
E-Mail: {hamideh, ammar, bob}@wins.uva.nl

Abstract

Water supply industries nowadays lack a global overview of the status of the production and the water distribution system. Distinct functionalities required in this industry, e.g. optimization, water quality, etc. are supported by independent, heterogeneous, and autonomous subsystems. Each subsystem performs its specific activity, but their co-working and complex information exchange needs to be properly supported. Typically, there is none or little coordinated control in order to assure a continuous supply, meet the quality standards, save energy, optimize pipeline sizes and reduce wastes. The development of the ESPRIT project WATERNET[1] (Knowledge Capture for Advanced Supervision of Water Distribution Network) involves the development of its different subsystems and the integration of these subsystems into a coherent environment, in which they can easily access and exchange the information they need. From the information management point of view, in order to support the requirements of advanced water production environment, there is a need to develop a strong interoperable information management system to support the cooperative heterogeneous subsystems with their exchange and handling of large amount of data. The PEER federated information management system developed at the University of Amsterdam is used for the development of the Distributed Information Management System (DIMS) layer for every subsystem in the WATERNET environment.

This paper first briefly describes the WATERNET infrastructure and its main components and then addresses the architecture and mechanisms developed for the information integration in WATERNET system. Furthermore, the paper describes how the integration architecture supports the required openness, flexibility, and future expansion requirements for the water management systems. The design of the innovative integration architecture, described in the paper, is generalized enough to be applied to other complex application environments, that involve the interoperability among heterogeneous and autonomous subsystems.

Keywords

Co-operative Databases, Modelling Multi-Agent Environments, Water Management Networks, Federated/Distributed databases, Open and flexible integration of subsystems

[1] The work described here has been partially supported by the ESPRIT project 22.186 - WATERNET whose partners are: ESTEC, UNINOVA, SEBETIA, WBE, Univ. of Amsterdam, ADASA, Smas-Sintra, Univ. Politecnica de Catalunya, ALFAMICRO, and Univ. of Naples.

1. Introduction

Water industries today require the cooperation of heterogeneous subsystems (also called units in this paper). Each subsystem, e.g. optimization, water quality, simulation, supervision, etc. performs a distinct function and their co-working and information exchange are of complicated nature. In principle, a number of activities may be assumed by every subsystem. Clearly, the number of units and the complexity of every system depend on the size and functionalities of the water industry. In Europe, water industries constitute a wide range, for example as small as a company where all modules run on a single system that is located in the control room of its head quarter, or as large as a water company with many geographically distributed control, processing, and distribution sites.

Independent of the size, water companies today *lack a global overview* of the status of production and of the water distribution network. Control of such systems, is often carried out locally, based on the operators' experiences. Typically, there is none or little coordinated control, that is needed to assure a continuous supply, meet the quality standards, save energy, optimize pipeline sizes and reduce wastes [3].

The main focus of the WATERNET project is two-fold: (1) the development of several subsystems performing the necessary functionalities (i.e. the supervision, the simulation, the machine learning, the models manager, the optimization, the remote unit, and the water quality); and (2) the integration of these subsystems into a coherent environment, in which the subsystems can easily access and exchange the information they need from the other subsystems, in order to function properly. The integration/interoperation architecture designed for WATERNET, involves the development of Distributed Information Management system (DIMS) for every subsystem, that provides all mechanisms necessary for such interactions among the subsystems. Considering the fact that the DIMS plays the role of the interlocutor/integrator among all other subsystems, its implementation must reflect the inter-operation requirements specific to the design of WATERNET system and its subsystems.

Here, the two main requirements to be considered (also described below) involve:

1) The need for provision of the information produced by every subsystem, for access by any other subsystem.
2) The general requirements of “openness and flexibility” to support the WATERNET system life cycle.

Subsystems in cooperative environment are independent and autonomous modules developed by different individual partners within the community. The best approach to support point (1) above, while preserving subsystem autonomy, is the federated database approach. The PEER federated database system is used for the integration of subsystems’ information through their DIMSs, and properly supports this point. The federated schema management of PEER [11] employs a common means for information representation; namely, a common object-oriented schema representation that acts as the “mediator” representation of all existing information within the subsystems.

In order to support point (2) of openness above however, in addition to the federated information management, we have chosen an integration mechanism and approach, that develops “Adapters” for every subsystem. The primary role of Data Adapters is to provide specific interfaces for the input/output data used by every subsystem program from/to the information representation in the common mediator schema in its DIMS. This mechanism in turn supports the openness of the system as an environment to which different functionalities can be simply added or removed, as required by any cooperative environment, in order to adjust to its specific needs.

The remainder of this document is organized as follow. Section 2 addresses the description and analysis of the water management system as well as the design of the federated architecture for the water supply network. A general and open implementation framework for the distributed WATERNET system is described in section 3. This framework includes a brief description of the PEER distributed/federated system developed at the University of Amsterdam. Section 4 describes the main integration architecture of WATERNET supported by the PEER federated system, in which the information sharing and data exchange is supported through the integrated schema. In section 5 an extensible integration approach supporting systems flexibility and application modularity through the use of the adapter component is presented. Section 6 enumerates the major characteristics and benefits of the extended federated integration/interoperation approach for the WATERNET system. Finally, section 7 concludes the paper.

2. Water Network Management

In water supply and distribution network, typically the information about the water characteristics and network devices, is gathered in remote units and processed at different stages of network simulation, network behavior learning, strategy optimization, and water quality checking. Furthermore, the proper planing and strategies for water management and processing untreated water are achieved under the supervision of the system supervisor.

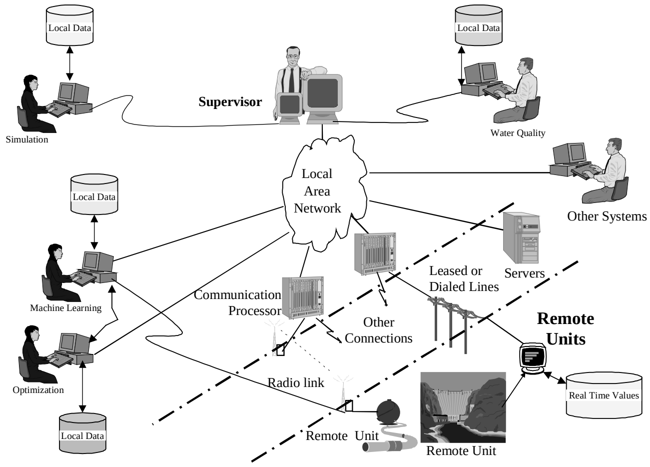

Figure 1: Water Management Environment

As depicted on figure 1, existing systems for control and monitoring of water production and distribution are heterogeneous and of different levels of automation and reliability. In such a cooperative environment, the proper functionality of all subsystems involved in the water control system depends on the *sharing* and *exchange* of data with other subsystems. Typically, direct connection between these subsystems is none existent or at best exists as point-to-point, where the exchanged information for example consists of documents, phone calls, and electronic mails. Some earlier publications address the problem of developing an infrastructure and/or mechanisms to support the systematic sharing and exchange of information [3,4,5], but the suggested solutions still lack a coherent environment to provide a global overview of the status of water production and the water distribution, an integration strategy for the considered subsystems and their activities, and support for the "openness and flexibility" requirements [10].

Following are the different kinds of subsystems that are considered necessary for the development of the WATERNET system [2]:

1- **Remote Unit subsystem:** remote unit represents the concept of a site where the information is gathered from a set of sensors and control devices, and some local control is executed. Every remote unit keeps track of the local information of the site (basically device information, status readings, alarm events, and commands) and is able to handle some local events by itself

2- **Supervisory subsystem:** supervisory element performs some central supervision and control of the water supply and distribution system. In some cases, there could be only one supervisory subsystem in the network, but then some level of fault-tolerance needs to be implemented. However, it is also possible to have multiple supervisory systems distributed along the network. Usually, under the normal conditions, only the supervisory system makes the final control decisions in the system to modify the behavior of a Remote Unit. In this case, the other units (e.g. simulation, optimization, etc. described below) can only ‘‘suggest” certain actions to the supervisory system, but the latter will make the final decision at the end. The functionalities of the supervisory system include:

a) **Planning**: planning is a daily process that allows a supervisor at the control head quarter to recognize failures in the system, to identify the non-optimized operations and resource wastes, and to take the proper recovery and maintenance actions. In order to achieve a good water supply, two sets of results from subsystems described below can be used: the water demand forecast generated by the Machine Learning subsystem and the optimized strategy proposed by the optimization subsystem.
b) **Controlling:** controlling is the main regulatory task of the water management network, it focuses on three operations: choosing a plan and making it the practical strategy for remote units, manipulating the devices, and adjustment of set points.
c) **Monitoring**: monitoring is an important process in water distribution network since it watches at the system by reading values of all devices and checks if everything runs properly (parameters should be within the defined range limits, default values, as well as rules checking for pressure, level, flow, etc.). The monitoring system can monitor all remote units for the company regarding their actual running status and collected information, periodic readings of network devices, alarms, and the sensor values about the pressure, flow, and quality for all remote units. The monitoring interface supports the browsing of Network Current Status, Historical Data, and Graphic Display of the device information and statistical analysis.
d) **Alarms Handling**: once an alarm is detected it will be presented to the system supervisor at the control room, the supervisor shall then make an expert decision on how to react properly to the alarm. In this case the knowledge/rules extracted by the Machine Learning subsystem (described below) can be used in order to help the supervisor to react properly, taking the suggested rules into consideration [7].

3- **Simulation subsystem:** simulation assists the operator if he/she decides to look forward in time (e.g. for a few hours) to spot potential problems that can develop if the network is not monitored aggressively. Finding such problems can be supported through the use of the most up-to-date consumption forecasts for the network. In this case, the simulation process looks at what will happen during the next hours, with the goal to spot the eventual problems before they actually develop and occur, a set of simulated network results will be produced by this subsystem and presented to the supervisory system

4- **Machine Learning subsystem:** that complements the work of the supervisory subsystem by other functionalities [6]. Two activities are supported by the machine learning subsystem.
   a) The *knowledge extractor* process that uses the network model information and the historical data for knowledge and rules extraction in order to be used by the supervision system for an advanced monitoring of the system.
   b) The *water demand forecasting* process for which the necessity resides in the idea of getting an overview on how operations are foreseen in future (e.g. tomorrow), so the system supervisor can have some knowledge about the problems expected, the possibility of changing the water supply because of maintenance, the optimal plan in terms of energy costs, and so on.

5- **Optimization subsystem**: optimization in the water networks refers to the computation of set points for the elements controlling water transfer in the network such as the pumps or valves related to cost, quality, etc. The computation is based on the forecast of future demands over the time horizon using a simplified model of the water network’s dynamic behavior.

6- **Water Quality subsystem**: quality in the water management comprises a large set of parameters, however the most important are related to the quality of supply (pressure, flow, continuity, etc.) and biological characteristics. The quality monitoring process gets actual values of the sensors for quality measurement from the supervision system and then it generates a list of possible abnormal situations for which a set of alarms will be generated and presented to the supervisor at the control room.

7- **Other External Subsystems:** some external subsystems may run outside the water management system, while they are needed to be contacted in order to provide information/services necessary for water distribution. For instance, the geographic information systems and/or the water network maintenance systems that can be contacted by the supervisory subsystem or others, when their information/services are required.

2.1. Water Network Analysis

As described above, subsystems involved in water management systems, are heterogeneous and geographically distributed. Through the analysis of the existing water supply and management networks [2], we have identified and classified the heterogeneous and distributed water management subsystems into four categories of units, namely: *control* unit, *remote* unit, *auxiliary* unit, and *external* unit (figure 2):

* The *Control Unit* performs some central supervision and control of the water supply and distribution system. Usually, under the normal conditions, the Auxiliary Units can only ``suggest" certain actions to

the Control Unit, but the latter will make the final decision.

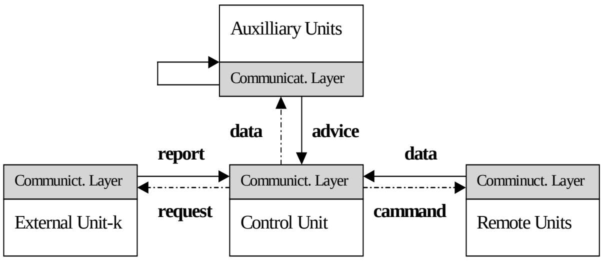

Figure 2: Logical units for the system architecture

* The *Remote Unit* represents the concept of a site where the information is gathered from a set of sensors and control devices.

* The *Auxiliary Unit* is a unit that complements the work of the Control Unit with other functionalities. Examples of Auxiliary Units include the machine learning unit, simulation unit, optimization unit, and water quality monitoring unit. These units will read the information from the Remote Units and/or the Control Units, and give the proper feedback in terms of certain suggestive actions (commands or parameter modifications) in order to achieve a better performance.

* The *External Unit* is a unit that typically functions outside the water management system, but needs to be accessed to provide some information and/or services. Examples of external units can include: the geographic information systems that could be provided by some services provider, the companies that can perform water network device maintenance, and the centres that can collect users complaints.

2.2. Information Management Approach

Units involved in the water control network (e.g. supervision, simulation, optimization, machine learning, and water quality) function properly if and only if they can access the information produced by other units. Therefore, the sharing and exchange of information among subsystems must be properly supported, while the proper independence and autonomy of the units needs to be also preserved. For instance, the control unit or an external unit are autonomous units, while the remote unit has only partial control over its functionality and takes orders from the Control Unit. Similarly, the heterogeneity of information representation in different units and its varied classification needs to be supported. In general, the same piece of information is viewed differently by two units, and different levels of details can be associated with it [3,8,9].

In order to support the complex information management requirements in water environments and applications, we have designed a comprehensive architecture for the WATERNET system (figure 3), in which every subsystem is augmented with a DIMS (Distributed Information Management System) layer that ensures the access in run time (via remote queries) to information stored in other subsystems. In general, subsystems are independent and self-serving, with a large variety of data that they generate and store. Therefore, any assumption of centralization, replication, or unification of data descriptions in different subsystems (through one global schema) is unrealistic. Therefore, it is preferred to have no centralized global schema or redundant storage of data within the entire network.

2.3. Simple Scenario for Subsystems interaction

The WATERNET system operation requires a real cooperative environment in terms of the integration and the exchange of information between different subsystems. In order to give the reader an overview on how complicated is the interaction between the WATERNET subsystems for data sharing and some results validation, a simple scenario for the process involved in developing an optimized strategy is presented in this section.

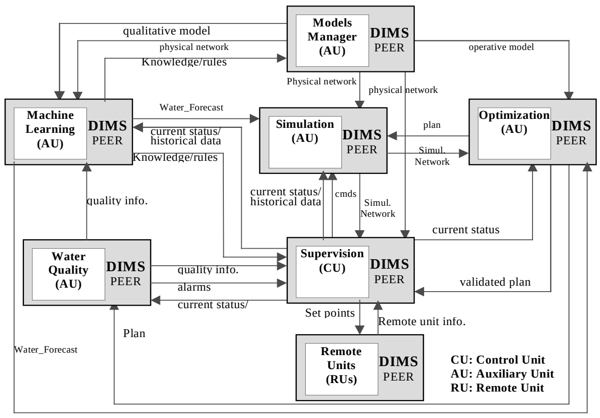

Figure 3: The water network architecture in terms of units

The cooperative work required to develop an optimized strategy can be considered as *"a part of the bigger cooperative environment required every day*", to identify many operations to be carried out the next day. The cooperative process needed for the simple scenario involves the optimization, machine learning, simulation, water quality, and the supervision subsystems.

Steps involved in the scenario:

- **First**, in order to generate a *management plan* for the next day operation of the network, the optimization asks the supervision for the network devices information; asks the machine learning for the forecasting results; and asks the models manager for the operative model. The management plan generated by the optimization subsystem, primarily consists of a sequence of commands to be performed at specific times on the network, e.g. opening a valve at 2 AM, stopping a pump at 5 PM, and so on.
- **Second**, the simulation and the water quality subsystems are invoked by the optimization. These two subsystems must access the *generated plan* information

from the optimization subsystem, perform some processing and give their feedback about the consequences of the *generated plan* on the ability of the system to support the proposed plan and/or how this plan affects the quality of the water.

- **Third**, the optimization subsystem needs to access both the simulation and the water quality subsystems, in order to check their evaluation results of its earlier generated plan and in order to decide either to recommend the plan to the supervision as an optimized plan or to reject it. If the plan is rejected, the whole process described above needs to be restarted to develop and test a new plan. Otherwise, the plan will be approved as an optimized plan and presented to the supervisor for acceptance. If accepted by the supervisor, the plan will be loaded at the remote units by the system supervisor at the control room; otherwise it will be canceled and re-planing starts again.

3. The WATERNET System Implementation

The architecture designed for the WATERNET system is comprehensive enough to support different possible implementation strategies adopted in water companies. Namely, it can support a wide range of companies. For instance, it can support on one hand the case of a small water company where all modules of the WATERNET system run on a single system in the control room at the head quarter, and the remote units only send their collected data to this head quarter. At the same time, it can support a medium to large size water company with many geographically distributed control sites, even if different modules of the WATERNET system, for instance, the forecasting, machine learning, and water quality management each run on different sites and are connected only through the communication network. The PEER federated database system [11], developed at the University of Amsterdam in C under Unix operating system, is used as the base for the implementation of the information management in the WATERNET project and supports the communication and interoperability among these subsystems. However, the PEER system was extended to better adjust to both to the specificities of the water management environment and the specific development strategies of different subsystems in WATERNET.
Some extensions enhanced the portability of the PEER federated system. For example, the development of two interfaces: the *on-line PC interface* and the *programming languages interface* for PEER. Considering the facts that PEER is Unix based, while most WATERNET subsystems are developed and run on PCs, the on-line PC interface developed for PEER, allows a user to interact with any database within the cooperative community in order to check, retrieve or update the information for which he/she has gained the appropriate access rights. The Programming Languages Interface includes the necessary set of functions to allow programmers to develop their own programs, while interfacing with PEER through several different applications programs written in C, C++, Pascal, Delphi, etc.
In the general architecture (as presented in Figure 3), every component of the WATERNET system being a remote unit, a control unit, an auxiliary unit, or an external unit, constitutes a PEER node. In principal, one unit can either run on an individual workstation (or PC), or several units can run on the same system. The PEER system and the development of the PEER federated layer for DIMSs are further described in this section.

3.1 The PEER Federated Layer

The PEER system provides an environment for the cooperation and information exchange among different nodes in a network, where every node is composed of one server process and may consist of several client processes. The federated schema management and the federated query processing of PEER (Figure 4) support the sharing and exchange of information among nodes, without the need for data redundancy and/or creation of one global schema. Therefore, the problems of data consistency, referential integrity and update propagation, and the need for a common glossary of concepts and definitions are eliminated.

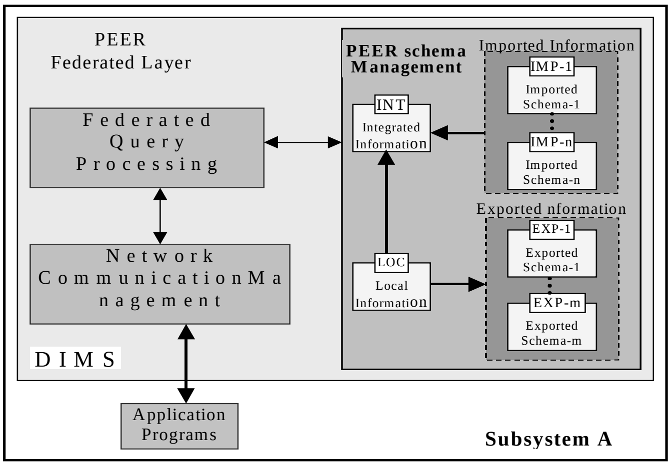

Figure 4: **PEER Federated Layer representation**

The federated schema management of PEER organizes four different kinds of schemas for every node (see figure 4). The local data at the node is defined by the schema called LOC. Every node can create other several schemas called exported schemas (EXP), to represent a part of its local schema (LOC); and only authorized users at remote nodes can access data of this node through some EXP schemas. The authorized nodes can import the EXPs of other nodes that will be called imported schemas (IMP). The imported schemas (IMPs) are then merged with the LOC to build the integrated schema (INT) for the node. Hence, every node in the federated community can access both its local and the remote information (from other nodes) through its INT schema, as if all the data is local information. At the same time, the physical and logical distribution of information becomes completely transparent to the users. The four kinds of schemas for the subsystems are defined using the SDDL (Schema Definition and Derivation Language) of PEER [11]. Several examples of these schemas defined for the WATERNET subsystems are included and described in earlier publications [1,2].

4. Base Integration Approach

An important outcome of the DIMS integration approach is that any subsystem in WATERNET can develop its application programs *without* the need of knowledge about the *format, structure*, and/or *location* of the data produced somewhere else in another subsystem.

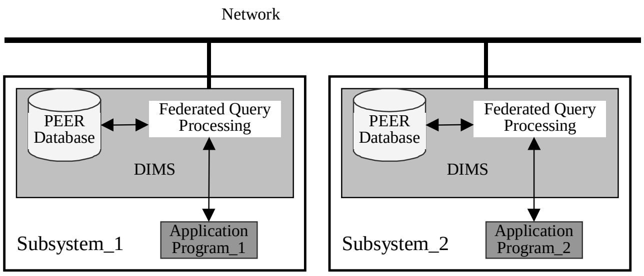

Figure 5 the Integration architecture

Figure 5 represents the main integration architecture of WATERNET based on PEER, in which each subsystem within the WATERNET is augmented with its DIMS (a PEER federated layer). Within the PEER federated database environment, the information sharing and data exchange is supported through the integrated schemas. Using the provided mechanisms, users and application programs in a subsystem can specify the queries for data retrieval or data insertion through the subsystem's integrated schemas. The defined queries can be specified on line by human users, using the on-line PC-interface, or within an application program using the programming languages-interface. Once a query arrives, based on the definitions in the integrated schema of the PEER federated layer, the query is decomposed into several sub-queries. Each sub-query is then sent to the proper remote subsystem. Finally, the local partial-result for the query is merged with the remote partial-results, and produces the complete result to the query, that will be presented to the end-user and appears just the same as if it was handled completely local at this node.

The approach described above allows a complete integration of the data stored in different subsystems in a transparent way, while preserving access security issues, and the execution of concurrent transactions.

5. Extended Integration Approach

As represented in figure 5, an application program within a subsystem can then simply receive its input from (and similarly send its output to) the PEER database server. However, considering the specific characteristics that define every application domain, this implementation architecture may not sufficiently support all the requirements within the WATERNET environment. In specific, to support the water management applications and the wide variety of subsystems within the configuration of different water companies, in our requirement analysis stage we have identified the need for a more "open and flexible" architecture. For instance, from time to time different subsystems (mostly pre-existing and some commercial, e.g. new simulation software) may need to be added to (or removed from) the WATERNET system, in order to better support the specific needs of the company. Even as a product, some of the existing subsystems may need to be disconnected from WATERNET, and/or replaced by other existing or new commercial products that run in the company. Using the federated architecture and approach as described above, these alterations within the subsystems require that subsystem developers have database language expertise to properly add/remove/replace the subsystems to the federated architecture. For instance, the knowledge of PEER database language commands [2] is mandatory, to generate the appropriate PEER commands to be included within any application program written in a subsystem in order to develop the interaction between a new subsystem and its DIMS. A similar problem arises when a unit decides to use as input (for its application programs) some other resources outside the DIMSs that may be available from external applications or databases. Similarly, when a subsystem wishes to generate the output of its application programs in a format that it can be also sent to other external applications outside WATERNET. Hence, there is a need for an open and flexible integration architecture. Under the influence of this "openness" requirement, we have extended the integration architecture of the DIMS to also include the "***Adapter***" (or data adapters) components, described in section 5.1.

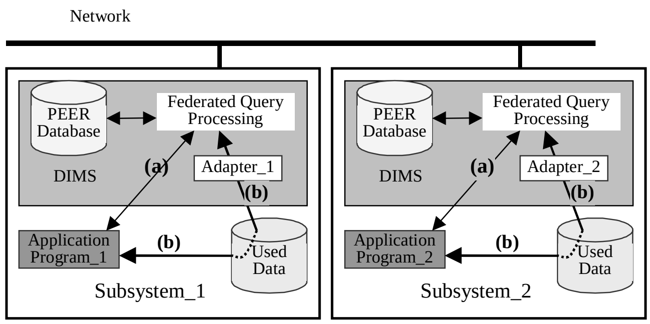

Figure 6: **extended integration architecture**

This extensible integration approach supports the system flexibility and application programs modularity for the WATERNET subsystems. The extended approach, as depicted in Figure 6: through (a) preserves the main properties for cooperative working in multi-agent environment, such as the data-location transparency, access security, transactions concurrency, etc. (similar to the architecture described in section 4), but additionally through (b) with the **adapters,** supports the openness requirement. Among other features, the adapters support adding/removing new subsystems within the WATERNET system that can be developed independently from the WATERNET project. Using the adapters, an application can receive its input either from the remote DIMS or from external application. Similarly, the generated output (in addition to storing it in DIMS) if needed can also be stored locally in a storage facility (or another simple database system) and made available to external applications that may not even be allowed to access and retrieve information stored in different WATERNET DIMSs. Clearly, within the WATERNET system, the data of a subsystem stored within its DIMS can always be accessed by other WATERNET subsystems through the DIMS to DIMS interconnections.

5.1. Data Adapters Supporting Openness

The Adapter framework, as represented in figure 7, provides flexibility and openness, and facilities of convenience for the development of application programs. In other words, the programs can read/write their data in the most convenient way to them. For every subsystem the adapters constitute a set of dual *pre-processor* and *post-processor* components, where each pair supports the input/output of one of its application programs.
The adapter framework supports the following:

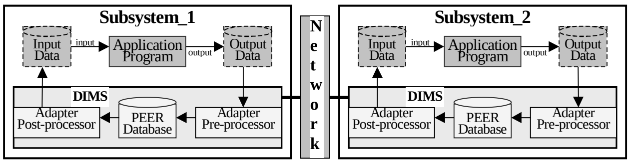

Figure 7: PEER layer – Federated data process using adapters

a) Provides the storage of the *exact output* of the application programs in every subsystem within its PEER database (DIMS layer). In fact, a module called ***pre-processor*** takes the output of every subsystem's program in the exact format that it is produced, (being a set of values, or a record, etc.), reformats it according to the object definitions in the subsystem's LOC schema and stores it in the DIMS.
b) Supports every subsystem's (e.g. supervision's) access to the data stored in other subsystems (e.g. optimization, machine learning, and others) in the *exact format* that is required by every application program. In fact, a module called ***post-processor*** provides access to imported data through the DIMS for every application program but changing its format to the exact format as desired to be read by the required application program (e.g. supervision's).

Therefore, the ***pre-processor*** and ***post-processor*** (in Figure 7) together provide the access *to* and *from* the PEER database for every application program in every subsystem. This mechanism in turn supports the modularity and autonomy of nodes within the cooperative community, while also supporting their desired specific application-program-dependent input/output formats for data.
Considering the above clarification, the DIMS integration architecture makes the development process of every subsystem (as well as adjustment to other environment configurations) very convenient, as it proved itself in practice during the development phase in the WATERNET project. Namely, every subsystem developed its application programs completely independent of the others, and it was enough to just specify to the DIMS developer the desired format for the *input* and *output* of those programs and not being concerned with how this data is produced by others. For instance, a program in machine learning subsystem produces as output "*a file*" for which the record format represents: "r_1, r_2, r_3, r_4, r_5, r_6". Meanwhile, for instance, a program in the simulation subsystem, reads its input from "*a file*" with the record format: "r_3, r_5, n_7, r_2, r_1, d_1", while here the "r_s" need to be imported from the DIMS of the machine learning subsystem, "n_7" needs to be imported from optimization subsystem, and "d_1" is a computation result using different imported and local values. At the last stage the imported information and other values need to be re-arranged according to the record format required by the simulation application program.

6. Major Characteristics and Benefits of Federated Integration/Interoperation for WATERNET

The federated schema management and federated query processing mechanisms of PEER, in addition to the adapters framework presented in the sections above provide a flexible, open, and reliable environment for the development of a strong water management system. Following characteristics resulted in this environment represent the major benefits gained from the approach taken in the project for the design and implementation of the DIMS.

1- System openness, so that different modules can be added to/removed from the WATERNET system, as needed, in order to support the specifities of different water companies. This characteristic strongly supports WATERNET as a flexible product, since in order to install the WATERNET system in a company, some of its subsystems may need to be disconnected from this product, and/or replaced by other existing products that already run in the company.
2- No need for the development of a single global schema (being centralized or distributed). As a result, there was no need for the development of a common glossary of concepts and definitions.
3- No need for data redundancy/duplication among the subsystems (no data transmission unless needed). As a result, the problems of data consistency, referential integrity, and update propagations are eliminated.

4- Complete transparency of logical/physical distribution of information among the nodes in the network, to the end user.
5- Retrieved data is always accessed directly from its origin and as a result it is always up to date.
6- The WATERNET development environment has become totally flexible. In fact, all subsystems continued developing *their functionalities and application programs*, while simultaneously the gradual and dynamic development of the DIMS adjusted itself to their extensions and modifications.

7. Conclusion

In this paper, a general methodology for the design of an open and flexible architecture for the integration between different WATERNET system units, and the mechanisms used for implementing the WATERNET framework were presented. The implementation of the designed architecture for the WATERNET framework is based on the PEER federated information management system, since it properly supports the cooperation and information exchange among different nodes involved in an intelligent cooperative environment. Within the current implementation architecture of the DIMS system, a small company can be properly supported, where all the modules run on a single system. At the same time, medium to large size companies with many geographically distributed control sites can also be supported. However, to better support the "openness and flexible" requirements in water management environments, the implementation architecture of the DIMS was extended to include the adapter framework. In addition to the main properties provided by the PEER federated layer in the DIMS implementation, the extensions with adapters provide among other features: (1) support for the systems expansion (addition, removal, or replacement of subsystems), (2) the adjustment to subsystems evolution (new/modified application programs), (3) the use of external media (resources from external application) as the input information, and (4) the storage of generated output in a different media, in order to be made available to external applications that may not even be allowed to access the information stored in different DIMSs within the community. The designed architecture and the implemented approach described in this paper can be applied to any other cooperative environment, in which several heterogeneous nodes need to interact and exchange their information.

8. References

[1] H. Afsarmanesh, A. Benabdelkader, and L.O. Hertzberger 1998. Cooperative Information Management for Distributed Production Nodes. *In Proceedings of the 10th IFIP International Conference PROLAMAT '98*, Trento, Italy: Chapman & Hall Press.

[2] A. Benabdelkader, H. Afsarmanesh, Cesar Garita 1998. Specification of an Object-Oriented Infrastructure for Data Management in Water Industries. Technical Report CS-98-06 (WATERNET Esprit Project 22.186), Department of Computer Science, University of Amsterdam.

[3] B. Ulanicki, J. P. Rance, P. L. M. Bounds 1997. A systematic approach to Information integration in water company. *International Conference on Computing and Control for the Water Industry*.

[4] C. G. Wang 1997. Object-oriented modeling for the operational control of water distribution system. *International Conference on Computing and Control for the Water Industry*.

[5] G. Cembrano, J.Quevedo 97. Optimization in water networks. *The international Conference on Computing and Control for the Water Industry*.

[6] Luis Manuel Camarinha-Matos, Fernando Martinelli (1997-a) Application of Machine Learning in Water Distribution Networks: An Initial Study, Workshop on Machine Learning Application in the real world; Methodological Aspects and Implications, Nashville, USA, July, 1997.

[7] H. Afsarmanesh, F. Tuijnman, M. Wiedijk, L.O. Hertzberger 1993. Distributed Schema Management in a Cooperation Network of Autonomous Agents. *In Proceeding of the 4th International Conference on Database and Expert Systems Applications DEXA'93*, LNCS 720: Springer-Verlag Press.

[8] M. P. Reddy, B. E. Prasad, P. G. Reddy, and Amar Gupta 1994. A methodology for Integration of Heterogeneous Databases, *IEEE transactions on Knowledge and data Engineering*, vol. 6, no. 6, December 94.

[9] S. Spaccapietra, Christine Parent 1994. View Integration: A Step Forward in solving structural Conflics, *IEEE transactions on Knowledge and data Engineering*, vol. 6, no. 2, April 94.

[10] H. Afsarmanesh, M. Wiedijk, and L.O. Hertzberger 1994. Flexible and Dynamic Integration of Multiple Information Bases. *In Proceedings of the 5th IEEE International Conference on Database and Expert Systems Applications DEXA'94*, LNCS 856, 744-753. Athens, Greece: Springer Verlag Press.

[11] F. Tuijnman and H. Afsarmanesh 1993. Management of shared data in federated cooperative PEER environment. *International Journal of Intelligent and Cooperative Information Systems (IJICIS)*, 2(4):451-473.