Article:

Liu, R., Wei, J., Ren, Y. et al. (4 more authors) (2017) HydroMP – a computing platform for hydrodynamic simulation based on cloud computing. Journal of Hydroinformatics, 19 (5). jh2017140. ISSN 1464-7141

https://doi.org/10.2166/hydro.2017.140

HydroMP – A Computing Platform for Hydrodynamic Simulation based on Cloud Computing

Ronghua Liu[a,b], Jiahua Wei[b,c,d*], Yan Ren[b,d], Qi Liu[a,b], Guangqian Wang[b,c,d], Songdong Shao[c,d,e] and Shuang Tang[f]

[a] China Institute of Water Resources and Hydropower Research, Beijing 100038, China
[b] State Key Laboratory of Plateau Ecology and Agriculture, Qinghai University, Xining 810016, China
[c] State Key Laboratory of Hydroscience and Engineering, Tsinghua University, Beijing 100084, China
[d] School of Hydraulic and Electric Engineering, Qinghai University, Xining 810016, China
[e] Department of Civil and Structural Engineering, University of Sheffield, Sheffield S1 3JD, Unite Kingdom
[f] PetroChina Research Institute of Petroleum Exploration & Development (RIPED), Beijing 100083, China

ABSTRACT

Modern water management decisions are increasingly dependent on efficient numerical simulations of multiple scenarios with multi-models. In this paper, a service mode for the hydrodynamic simulation based on cloud computing is proposed, and the relevant frameworks of the Hydrologic/Hydraulic Modeling Platform (HydroMP) are designed and implemented. Various hydro-models can be integrated into HydroMP dynamically without the need of program recompiling, since it achieves the scheduling of computing resources to provide end users with the rapid computing capacity of concurrent scenario simulations in the form of a Web service. The present study focuses on the dynamic model integration, resource scheduling, system communication and data structure design, and achieved the following four objectives: (1) Two model integration approaches, including executable program file (EXE) mode and program interactive integration mode (PIIM). The former is used to realize the rapid legacy model integration and the latter ensures real-time communications between the model and platform during computing process; (2) Based on the two-layer computing resource scheduling framework

---

[*] Corresponding author. Tel.: +86 (0)10 62796325. Fax: +86 (0)10 62782463
E-mail address: weijiahua@tsinghua.edu.cn (J. H. Wei)

and scheduling rule of the simulation scenario level, the self-adaptive scheduling and concurrent control of the cloud computing resources are proposed; (3) Through API provided by the Windows HPC Pack, a scalable integration of the platform, HPC Cluster, is established and the scenario data and hydro-models are integrated; (4) A web service is used to provide the hydraulic simulations, result feedbacks and other interfaces to fulfill the computing service. To use the present one-dimensional hydrodynamic cloud computing as a prototype, different integration methods will be applied to construct the CE-QUAL-RIV1 and JPWSPC (Joint Point Water Stage Prediction and Correction) models, thereby to investigate real-time scheduling of the water transfer channels in the South-to-North Water Diversion (SNWD) project. The results showed that massive modeling scenarios by use of different hydrodynamic models, if submitted concurrently, can be processed simultaneously in the HydroMP. The data structure of the proposed framework can also be extended to two-dimensional and three-dimensional hydrodynamic situations.

Keywords: cloud computing; hydrodynamic model; model integration; web service; HydroMP

1 BACKGROUND

Hydraulic calculation is an important aspect of the environmental simulations. With the development of numerical analysis techniques and hydraulic theories (Launder and Spalding, 1974), numerical simulations of the hydraulic systems have been rapidly developed and also widely applied, and are now playing an important role in the water management (Wei et al., 2008; Reed et al., 2007; Blöschl et al., 2008). As the requirement on elaboration and real-time in water management increases, hydraulic modeling is facing great changes and challenges. An elaborate simulation requires the high computing precision, as manifested by finer simulation granularity in a large system domain. Besides, a real-time requirement necessitates the rapid acquisition of simulation results of multiple scenarios for the decision-making. For example, in the event of sudden flooding or water pollution, multiple scheduling plans need to be simultaneously and rapidly considered to propose optimum solutions through comparisons (Zeng et al., 2010; Wu and Wang, 2010; Wang et al., 2009). Therefore, the trend of elaborate and real-time management raises the necessity of concurrent parallel running of multiple scenarios and rapid simulations during the decision-making. The multiple scenarios need to be computed concurrently in order to meet the demand of rapid simulations, especially the demand of rapid optimization during the water management decision-making. This poses a great challenge to the scalable computing resources, and associated efficient, reliable data communication mechanisms. Thus a new generation of modeling tools or services should be imperiously established, which allows for the integration of multi-models and scalable computing resources.

Over the past decades, especially with the publication and continuous development of the OpenMI standard of the EU Water Framework Directive (Gregersen et al., 2007), multi-model integrated systems have been developed rapidly. OpenMI provides a standard for coupling between multiple interdisciplinary models in step simulation, including model component interface specifications, definitions of data exchange objects, definitions of object linking and definitions of triggering methods. Several model integration systems based on OpenMI have been released, and HydroModeler is one typical application (Ames et al., 2012; Castronova et al., 2013). In HydroModeler modules, boundary data needed for the model component can be obtained by connecting DbReader components that meet the OpenMI specifications. Studies related to OpenMI specifications also included the integration of script models (Bulatewicz et al.,

2013) and evaluation on the performance of data exchanges between OpenMI-compliant components. Castronova and Goodall (2013) used OpenMI standards to split their rainfall-runoff model into three independent OpenMI-compatible model components forming a loosely coupled model system and performance tests showed the loosely coupled model did not affect the performance of the original system. In addition to integration platforms using the OpenMI standard, other integration platforms have also been built, e.g., the Object Modeling System (OMS), the Community Surface Dynamics Modeling System (CSDMS) and the Common Modeling Platform (CommonMP)[†]. The model developers who understood integration standards of the platform were able to implement the integration and reuse of the models by modifying only a small amount of code (David et al, 2013). CSDMS is a platform for model integration and sharing, which can provide high-performance computing capacity, and models integrated into this system can use its computing resources (Silva et al., 2012; Overeem et al., 2013). CommonMP is a platform that integrates multiple hydraulic and hydrology models; it manages the registration and use of these models via a model library. Schmitz et al (2014) developed an accumulator, a programmable general-purpose model building block executing custom scaling operations at model runtime, which can characterize runtime information of input and output variables required for the implementation of scaling operations between component models with different discretization. A processing conversion and parallel control platform (PCsP) is proposed for transitioning serial hydrodynamic simulators to a cluster-computing system (Shang et al., 2016).

These previous studies improved the overall level of model integration and simulation application services. However, all of the above integrated systems share a common feature, i.e.,

---

[†] http://framework.nilim.go.jp/en/index.html.

the source code must be modified in the model integration process, so as to rewrite the prototype engine into model components defined by the system. This would be difficult for a majority of legacy model developers, and thus to certain extent hinder the use and sharing of the legacy models. Meanwhile, existing studies have not focused on the rapid simulation of parallel multiple scenarios in a decision-making process, as well as the simulation as a service with the integration of computing resources. Although the CSDMS framework provides the HPC cluster, only model developers can use this feature; the use of the HPC cluster has not yet been converted into a modeling service that users can access via the Web. Model users and decision-makers are often unable to find solutions at computing resource bottlenecks to make the decision-making more efficient and obtain the most reliable and optimized outcome possible via concurrent massive scenarios. In particular, in a real-time hydraulic scenario optimized scheduling, typically hundreds of computing scenarios need to be simulated simultaneously, which poses a challenge to the computational capacity of the server. In the event that computing capacity struggles to meet the concurrent computing requirement, a newer, high-performance server needs to be purchased. This could cause a high investment and low usage. In a review on integrated environment modeling, Laniak et al. (2013) mentioned “modern and visionary work using concepts such as cloud-based computing and Web services to achieve a higher level of functionality in next generation IEM modeling frameworks,” and proposed that cloud computing and Web services are another important technique in addition to the model coupling. They pointed out that relative to traditional computing, the cloud computing has various advantages, such as saving cost, reducing development time, rapid integration of model components, and good capacity for simulation after the environment disintegration. The cloud computing thus dynamically packages the interactive services into a custom simulation system. In addition, cloud computing also promotes the development of a strong integration modeling community via service sharing such as through a Web service.

Regarding the advantages of cloud computing, in the last two years many scholars and relevant agencies have conducted studies on simulation services and integration platforms based on cloud computing. Sun (2013) used Environmental Decision Support System (EDSS) tools in the cloud services provided by Google Drive to solve high cost problems, to ease difficulty in information sharing and other problems in joint decision-making. Burger et al. (2012) applied the concept of cloud computing to integrate the computing power of supercomputer and hydrological models, and used GUI and other interfaces to provide the users with real-time simulation services of the ParFlow hydrological model. Lloyd et al. (2012) used Eucalyptus technology to build a virtual machine (VM)-based Cloud Services Innovation Platform (CSIP). Brooking and Hunter (2013) developed a Web-based repository to provide high-speed, interactive access to online simulations of hydrological models. Shi et al (2015) proposed a general framework for a service-oriented architecture (SOA) for ensemble flood forecast based on numerical weather prediction (NWP). Arango et al. (2014) developed a new version of Agent Swarm Optimization, taking advantages of the Cloud Service provided from the Windows Azure to support the analysis of a high number of scenarios. Glenis et al (2013) developed a parameter sweep version of the urban flood modelling, analysis and visualization software "CityCat". This can be deployed in a cloud environment to make use of the cloud computing resources, so as to be able to estimate the spatial and temporal flood risk at a whole city scale, which is much larger than what had previously been possible through using the cloud computing resources via the HTCondor. Besides, Rodriguez et al. (2014) developed a cloud-based Early Warning System (EWS) platform HIDROMET for real-time urban flood warning data, where the users can connect to the system through Internet by any device and the platform can integrate the user data and system data for warning analysis.

The above systems based on the cloud services are still under development or being improved. Some are able to implement the integration of particular models, and most focus on Web-based scenario searching and information presentation services. Based on the user demand for the integration of multiple models and for rapid concurrent computing, in the present study it is proposed to integrate hydraulic models with high-performance computing resources, and to provide modeling services using the computing power integrated in the system in the form of a Web service, i.e., Hydro-Modeling as a Service (HMaaS). HMaaS provides conceptual and technical support for model integration and the use of scalable computing resources from the idea of "scalability, automation, low-cost and efficient use of resources" in cloud computing (Hwang et al., 2011; Ari and Muhtaroglu, 2013; Gupta et al., 2013; Caballer et al., 2013; Huang et al., 2013). It achieves unlimited scalability of computing power with the idea of "Rapid elasticity" reduces cost with "customizable demand and usage-based billing” and implements streamlined and automated data management and computing services with the idea of a “resource pooling”. The users only need to learn a few methods for calling simulation interfaces or for using the terminal system, and do not need to understand the principles, implementation or the computing capacity of the cloud computing based modeling platform. In the present framework of web service and cloud computing, a cloud computing service platform for hydrodynamic simulation, HydroMP, has been developed. The objective of developing such a cloud computing based hydro-modelling platform is similar to that of the several others (Arango et al., 2014; Glenis et al., 2013; Rodriguez et al., 2014) in that all of them address the needs of multi-scenario and prompt feedback under highly intensive computing resources. On the other hand, most hydrodynamic and hydrological models are site-specific and they may not perform equally satisfactorily in other areas. Just to improve this situation, the cloud-computing based HydroMP platform aims to integrate the various conceptual and physical mechanisms together with the unified data structure and operational mode. Therefore, the users can freely select the most appropriate scenarios to run the models based on the available site information and the adaptability of each model to the region. The HydroMP platform system assumes its unique feature in the following three aspects: (1) Not only address the multi-scenario situation, HydroMP can also dynamically integrate different models and algorithms efficiently; (2) HydroMP adopts a two-layer structure framework for the system deployment and resource regulation, thus can have more potentials in its extendibility; and (3) HydroMP makes full use of the cloud computing resources through the constructed HPC clusters, therefore adopts a different way in exploring the cloud environment and its computing sources and technologies. The main characteristics of the HydroMP platform are as follows: (1) The hydro-models and the computing resources are highly integrated, and the unlimited computing power provided by distributed, scalable computing resources is used to meet the demands of multi-client, multi-scenario, simultaneous rapid simulations; (2) “Plug and play” integration between the platform and the models is achieved, and the provided universal model integration methods include all types of models, e.g. a large number of legacy models, so the users can choose the best adaptive one for a particular research region freely; (3) The platform provides scenario submission, progress inquiry, result feedback and other interfaces via Web services; (4) The platform provides open SDK (including data structure, method and interface), so that any client can reference the data structure and use the Web service to develop application systems (terminal) for a variety of purposes. In the application systems, there is no need for the computing power and hydro-models, because the system can call the interface via Web service to use these resources. In addition, HydroMP also employs a distributed computing framework, including a HydroMP center and some HydroMP servers. The HydroMP center and HydroMP servers can be deployed in distributed high-performance computing HPC clusters independently and the number of HydroMP servers is also extensible. The present HydroMP platform is very similar to other Software as a Service (SaaS) but it mainly serves the HPC computing clusters. Meanwhile, HydroMP is also a model integration system to provide different simulation models and algorithms to the user community. After multi-model integration, the unique advantage of HydroMP is that it can carry out a variety of model setups and solution algorithms based on only one dataset source.

To design and realize a cloud computing-based hydraulic modeling platform, the following technical issues need to be addressed: (1) fast, convenient, dynamic and standardized integration of the hydro-models; (2) employing, scheduling the computing resources; (3) data exchange and real-time communication between different programs and processes on the platform; (4) standardization of call interfaces and fast data transmission based on Web services. The present paper describes the details of key technologies to construct the framework of HydroMP. Subsequent parts of the paper are organized as follows: the next section is a general description of the platform and the framework of HydroMP server deployed in distributed HPC clusters; Section 3 describes the model integration method, resource scheduling, concurrent scenario simulation management, communication between multiple systems and multiple processes, implementation of Web service interfaces and other key technologies and methods used in the HydroMP platform; Section 4 describes the implementation of the HydroMP platform and the integration of 2 one-dimensional hydrodynamic models; Section 5 describes a practical case of the HydroMP application; the last section includes the conclusion and future prospects of HydroMP.

2 HydroMP FRAMEWORK

The HydroMP platform consists of a star-topology deployment structure, including a HydroMP center and a number of distributed HydroMP servers. Each HydroMP server is an individual computing service platform, and the HydroMP center is a load balancer between the HydroMP servers, as well as the entrance for terminals. The client can connect directly to the HydroMP servers, or connect to the HydroMP center and be forwarded to one HydroMP server via the balancer of the center. The deployment structure of the HydroMP platform is shown in Figure 1. In the lab test environment, three HPC clusters are used to deploy the computing services; one cluster is used as the HydroMP center, the second HPC cluster as a private HPC in Tsinghua University is used to deployed a HydroMP server, and the third HPC cluster was constructed in a commercial cloud computing platform Windows Azure in China, which is used to deploy a HydroMP server. During the running process, each HydroMP server dynamically sends the computing resource usage rate, the number of online users and status of simulating scenarios to the HydroMP center to analyze each server's load for granularity scheduling. Each HydroMP server can also be independently called by PC clients, mobile clients, tablet PC clients, through 12 service interfaces based on Web services. The HydroMP server can be registered to HydroMP center dynamically, and as the third HPC cluster, can be constructed in the cloud computing environment to provide elastic computing resources, so the HydroMP is supposed to a cloud computing based framework which can provide the elastic computing resources.

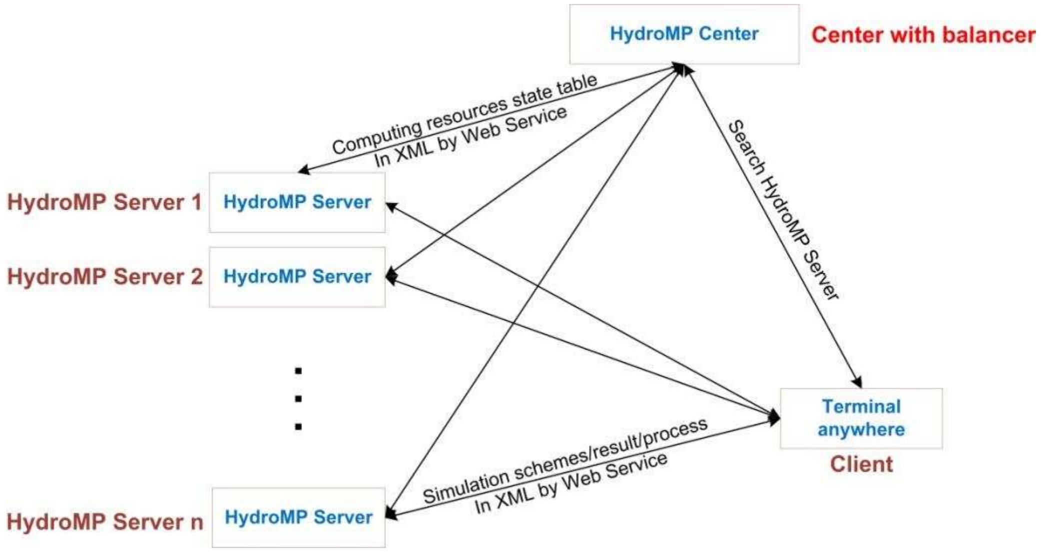


Fig. 1. Distribution of deployment of the HydroMP server

The HydroMP server in each cluster consists of five components: the management service system (A), the database system (B), the scheduler (C), the HPC cluster (D) and the hydro-models (E), as illustrated in Figure 2. Among these, the management service system is the general management platform and the window for providing services, including scenario management modules, computing management modules, model management modules and system management modules. It is responsible for the addition and deletion of scenario data, management of the model database, result inquiries and system management. All computing service interfaces need to be wrapped in the management service platform via a Web service. The management of all data, models and computing statuses on the HydroMP platform is also completed by the management service system. The HydroMP platform uses Oracle as the management tool for the unified management of basic data, scenario data, model registration and other information, and realizes the association between different data and union queries using a foreign key relationship. The scheduler is responsible for the startup, pause, restart, computing core allocation and computing workflow control, in order to achieve effective scheduling of computing resources. Windows HPC Server 2012 is used to manage the computing resources of the HPC Cluster, and the scheduler uses job management and task management APIs provided by the HPC server for job scheduling and computer core allocation. Hydro-models registered to the platform include executable programs and related DLL files, and the modeling programs used named pipes to communicate with the scheduler, including receiving the data, uploading progress and results.

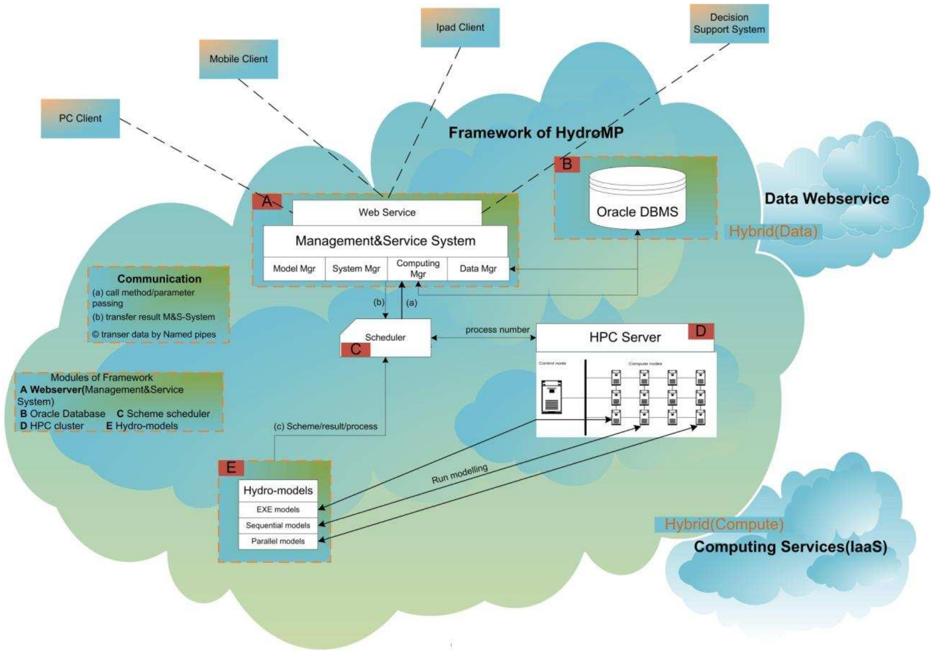


Fig. 2. Framework of HydroMP

3 MODEL INTEGRATION AND RESOURCE SCHEDULING IN HydroMP

HydroMP is constructed on a pool of HPC computing resources and hydro-models, so the platform needs to have the capacity to integrate and schedule the computing resources, and to have the ability to integrate multiple models. Some Web service interfaces to receive external calls are needed to provide the concurrent submissions and result queries of massive numbers of scenarios. As a multi-system and multi-process collaborative platform, the communication between various subsystems and processes is also very complex. To this end, the system must address four key issues: universal model integration, resource scheduling and concurrent processing, combined combinations between multiple systems and processes, and design of a Web service interface ("cloud - terminal" interaction).

3.1 Universal model integration methods

Existing model integration methods (e.g., OMS, CSDMS) requires the rewiring of the model code. Hence, it poses a challenge to the integration of legacy models. For this reason, the present study investigated a large number of legacy models, such as CE-QUAL series, Environmental Fluid Dynamics Code (EFDC), QUAK2K and some models currently used in Chinese research institutes. By taking into account the different characteristics of different models, HydroMP offers two model integration methods.

(1) EXE integration mode: Through analysis, it was found that the extent to which the inputs and outputs are structured in a large number of hydraulic simulation legacy models is relatively low, and typically the model developers customize the input and output file formats. The steps of model use are to first use a text editor (e.g., Notepad, Ultra Edit) to edit an input file needed for running the model, followed by starting the simulation; the model reads the input files during initialization, and after the completion of calculation it saves the results to certain output files using the customized format. Based on this, HydroMP uses loosely coupled communications between the platform and the models for integration. In this way the data I/O converter established between the platform and the models conducts the conversion, and the communication between the platform and the models is realized via size of output file for interaction. When the model is called, four steps (creating the input file, starting the simulation, reading the progress, and reading the result file) are executed by calling CreateInpFile(), ExecuteExeprogram(), GetComprocess() and ReadOutfile(), respectively, as shown in Figure 3. With this method there is no need to make any modifications on the code of the original models.

Instead, only the platform's software development kit (SDK) needs to be referenced and the corresponding interfaces need to be implemented, thereby achieving the “non-invasive” integration of the original models. EXE integration cannot realize the bidirectional model link between the different components while unidirectional model link is allowed through the platform operation. However, in practice it has been found that unidirectional model link is also quite common to address quite a few real needs. For example, most hydrodynamic and hydrological models fall into this category. By integrating the multi-models in the HydroMP platform, sensitivity analysis can be made on different numerical algorithms so as to eliminate the uncertainties of the simulation results.

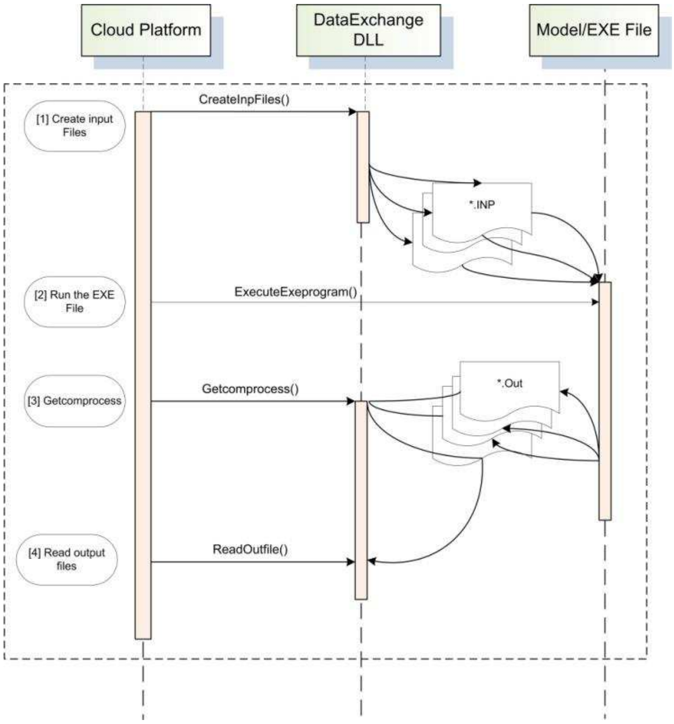


Fig. 3. Execution process and component interaction in EXE integration mode

(2) Program interactive integration mode (PIIM): To enable real-time interaction between a running model process and platform, the present study proposes PIIM. PIIM develops a standardized model wrapper program for communication between model process and the

computing scheduler of platform via named pipes, and uses five standardized communication interfaces to implement communication. The types of communication include the initialization, model execution, model pause, progress acquisition and results reading. The simulation platform sends all requests and the standardized wrapper program responses to the requests, and then returns the corresponding data, thereby realizing the control of simulation workflow by the platform. Inside the wrapper program, model initialization, preprocessing, simulation startup, single-step boundary updates, single-step execution, calculation completion and other interfaces are used to call methods in the model components, as illustrated in Figure 4. In the model integration, the model needs to be rewritten into a model component Model.DLL that fits specifications similar to the OpenMI interface standard, and when the wrapper program calls the model component it is compiled to a new executable file. There exist some kinds of difference between the PIIM and OpenMI integrations in that: (1) PIIM integration is implemented through the HydroMP platform regulator to control the simulation progress via the Named Pipes, while in the OpenMI the linkable component uses the event mechanism to drive model integration; (2) Different data transfer modes are implemented in the two systems, since HydroMP uses the HPC cluster for multiple and large scenario simulation which requires the linkage models to run on different computing nodes So PIIM uses the dataflow to transfer information via the Named Pipes and the designed standard platform data structure is similar to that in OpenMI, while OpenMI exchanges the data directly among the different components using the pull-driven approach via the memory. In this sense, the promising feature of the PIIM integration is that it can conveniently achieve the complex multi-model and multi-scenario analysis, for example, while simulating a flood event in a catchment area, it needs the coupling of hydrological and hydrodynamic models but usually the end-users are not very familiar with the runoff and channel conditions. There are three different runoff engines including Xin'anjiang runoff model, Horton runoff model and SAC runoff model, and two flood routing models including the diffusion wave model based the routing and dynamic Muskingum model. Through the data exchange in HydroMP, totally six runoff-routing coupling models could be created on five computing processes. This kind of flexibility is one attractive feature of the HydroMP platform to meet its multi-purpose target. The model integration framework, converse relation between original model code, data exchange file (DLL file), standard wrapper and executive file (EXE file), and the data conversion between executive file and HydroMP is shown in Figure 5.

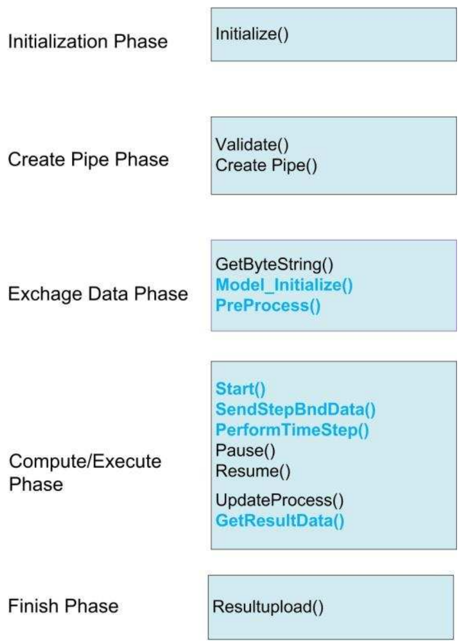


Fig. 4. Execution process in the PIIM integration mode

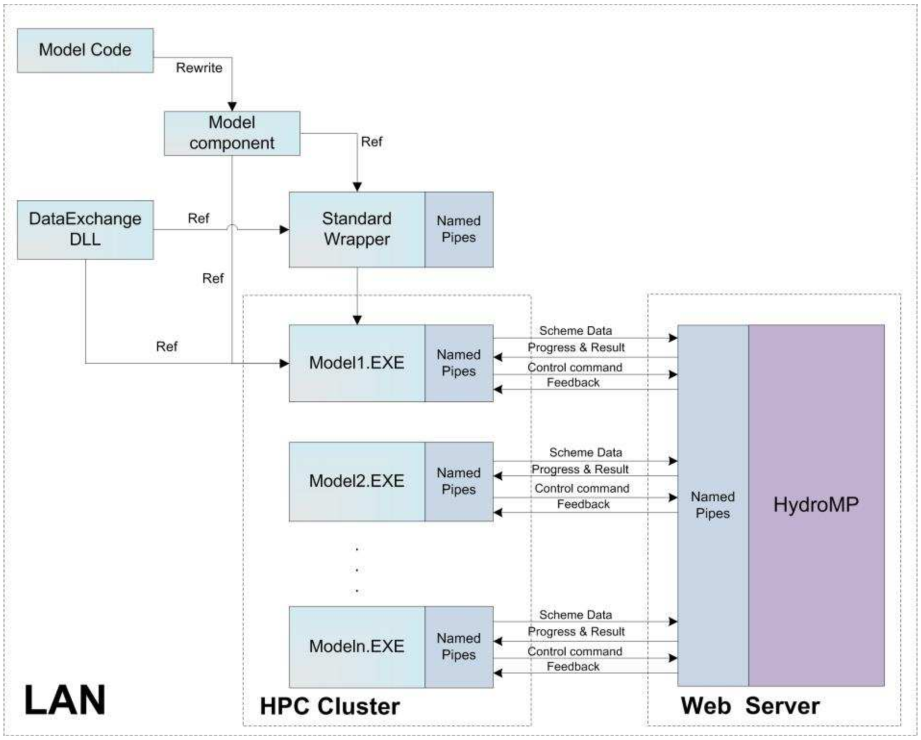


Fig. 5. The integration workflow and data conversion in PIIM mode

3.2 Computing resource scheduling

HydroMP is a distributed deployed simulation cloud platform with a HydroMP center and several HydroMP servers. The HydroMP server can be a standalone simulation platform as well as a branch center of HydroMP center. The HydroMP is a classic hybrid cloud computing framework. Each HydroMP server can provide partial computing resources to the public, and also be a computing platform for internal simulations as a private cloud computing service.

This design framework of HydroMP benefits the coupling of the public cloud computing and private one. The institute owning the computing resources can deploy a HydroMP server for the hydrodynamic simulation and rent some computing resources as a public cloud computing service when the computing resources are more than the needs of their institute, because the HydroMP center dynamically provides the HydroMP server registration. To a user who needs the computing resource can select the HydroMP center as well as any HydroMP server to drive the hydrodynamic simulation. The HydroMP center and HydroMP server can be deployed in a public cloud platform (i.e. The Amazon EC2 or Windows Azure) as well as in the private HPC clusters. The framework of the HydroMP and interactions between the center and servers are shown in Figure 6.

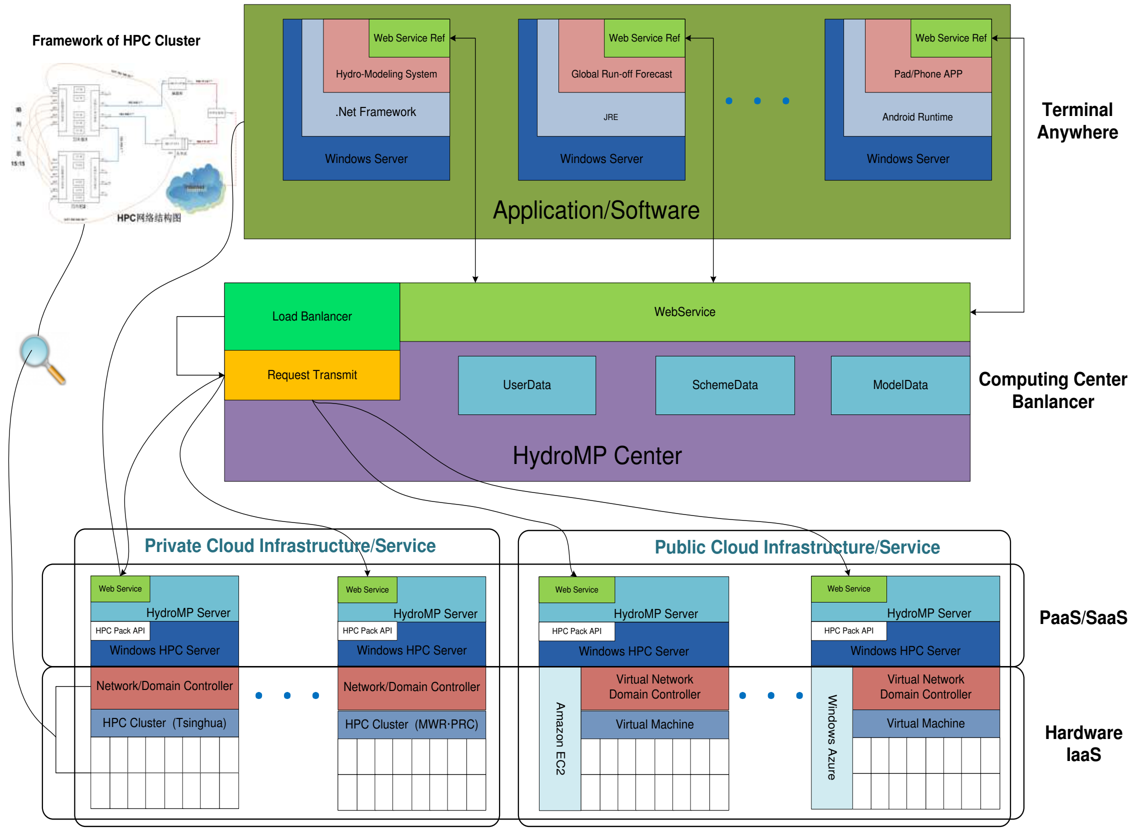
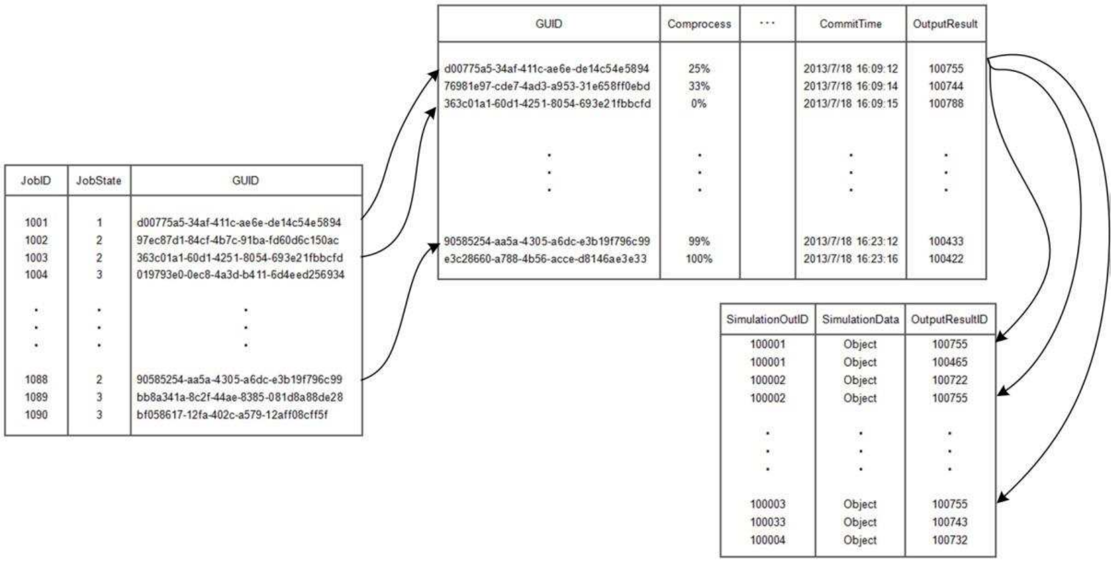


Fig. 6. The framework of HydroMP

The computing resources scheduling includes the balancing between HydroMP servers and the allocations in each individual HydroMP server. Firstly, the HydroMP center gets the available computing resources of each HydroMP server. Then it analyses the simulation tasks to evaluate the computing resource requirement and occupied time of each task. Lastly, the tasks are allocated to different HydroMP servers based on the load distribution matrix of all HydroMP servers using a particle swarm optimization algorithm. The objective of computing resources scheduling among all HydroMP servers is to reach the load balance between the HydroMP servers.

As an individual computing server, the HydroMP server will schedule the computing resources in their HPC servers after receiving the computing tasks from HydroMP center. A computing scheduling principle based on different levels of urgency and user is proposed in this paper. This scheduling principle is adapted for the needs of multi-users and different computing requirements. The core principle includes three aspects, the first one is that the urgent simulation task should be computed in real-time even if the other computing tasks need to be paused or stopped. The second one is that the task of paying users should be computing preferentially. The third one is that the intermediate results of paused tasks should be timely saved. In computing the resources scheduling, there are three different priority levels for computing the use: emergency computing, paying user computing and free user computing. The platform must ensure that the emergency computing is executed. When the unoccupied computing resources cannot meet the emergency computing scenario, some of the non-emergency computing should be paused. The computing scheduler is designed to implement the rules of resource scheduling. The schedulers are the bridge of resource allocations and data communications between the platform and the HPC management system.

In the present paper, a set of scenario computing status management tables have been designed to realize the sharing of computing status data among different sessions and different terminals. These tables include the Job-Scenario table, the Scenario-State table and Scenario-Result table, as shown in Figure 7. They record information of the association between the HPC and computing scenarios, the computing status and progress of the scenarios, and the results of the scenario calculation, respectively. The scheduler implements the resources allocation and scenario simulation progress management via a set of program flows including the new task create, status update, error report, results append, result storage, scenario resume, and task restart.



JobID	JobState	GUID
1001	1	d00775a5-34af-411c-ae6e-de14c54e5894
1002	2	97ec87d1-84cf-4b7c-91ba-fd60d6c150ac
1003	2	363c01a1-60d1-4251-8054-693e21fbbcfd
1004	3	019793e0-0ec8-4a3d-b411-6d4eed256934
⋮	⋮	⋮
1088	2	90585254-aa5a-4305-a6dc-e3b19f796c99
1089	3	bb8a341a-8c2f-44ae-8385-081d8a88de28
1090	3	bf058617-12fa-402c-a579-12aff08cff5f

GUID	Comprocess	···	CommitTime	OutputResult
d00775a5-34af-411c-ae6e-de14c54e5894	25%		2013/7/18 16:09:12	100755
76981e97-cde7-4ad3-a953-31e658ff0ebd	33%		2013/7/18 16:09:14	100744
363c01a1-60d1-4251-8054-693e21fbbcfd	0%		2013/7/18 16:09:15	100788
⋮	⋮		⋮	⋮
90585254-aa5a-4305-a6dc-e3b19f796c99	99%		2013/7/18 16:23:12	100433
e3c28660-a788-4b56-acce-d8146ae3e33	100%		2013/7/18 16:23:16	100422

SimulationOutID	SimulationData	OutputResultID
100001	Object	100755
100001	Object	100465
100002	Object	100722
100002	Object	100755
⋮	⋮	⋮
100003	Object	100755
100033	Object	100743
100004	Object	100732

Fig. 7. Scenario computing status management tables

3.3 The communication between multiple systems and processes

As shown in Figure 1, HydroMP can be divided into the management and service system, the HPC Server, the database system, the scheduler, and the model program. The "communication pairs" include: 1) communication between the management service system and the scheduler via the class referencing; 2) communication between the management service system and the database system via the Oracle data provider; 3) communication between the scheduler and the HPC clusters via the HPC computing resources scheduling interface; and 4) communication between the scheduler and the mathematical models via the Named pipes or in-out files.

3.4 Web service interface

HydroMP provides scenario calculations, progress inquiries and simulation results acquisition services through 12 Web service interfaces, including GetModels() for acquiring computing model information, ScenarioSubmit() for scenario submissions, GetState() for computing status queries, GetResult() for acquiring results, DownloadScenario() for downloading scenario data, DeleteScenario for deleting scenario, EditTimeSpan() for modifications of simulation timespan, and so on. Figure 8 illustrates the schematic process of user submission, status acquisition, real-time acquisition of the results and scenario deletion after acquisition of all results. The web service interface list and detailed information is shown in Table 1.

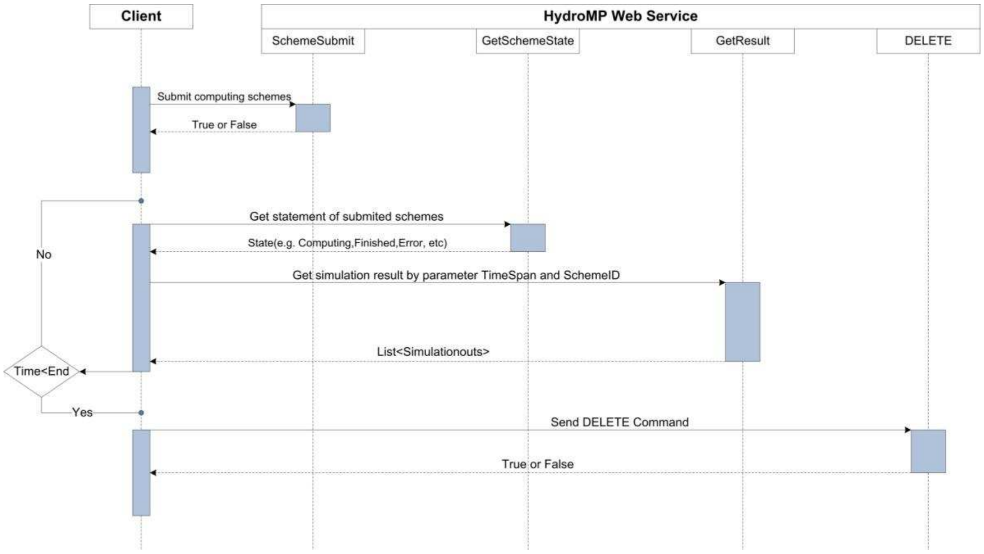


Fig. 8. Interactions between Client and HydroMP

Table1. Web Service List of HydroMP

No	**InterfaceName**	**Parameters**	**Comments**
1	UserLogin	UserName	To validate the log user
		Password	
		UserAddress	
2	GetModels	ModelType	To get the model list
3	ScenarioSubmit	ScenarioList	To submit the scenario or scenario list
		ModelID	

4	GetProgress	ScenarioID	To get the simulation progress of scenario
5	GetState	ScenarioID	To get the simulation state of scenario
6	GetSimulationOut	ScenarioID	To get the simulation results
7	EditTimeSpan	ScenarioID	To edit the simulation timespan of the scenario
		Start-Time	
		End-Time	
8	GetScenarioByID	ScenarioID	To get the scenario information by ScenarioID
9	DeleteScenario	ScenarioID	To delete the appointed scenario
10	GetResult	ScenarioID	To get the simulation result of scenario(s)
11	DelModel	ModelID	To delete a model engine
12	AddModel	ModelID	To register a model engine

4 PLATFORM IMPLEMENTATION

Currently, the HydroMP system provides a one-dimensional hydrodynamic library, the following describes the one-dimensional hydrodynamic modeling data structure design method and data hierarchy in HydroMP, the already integrated models in HydroMP and critical processes in integration, and the development method and functions of Cloud-Hydro1D (desktop software for one-dimensional hydrodynamic modeling and cloud computing established by using SDK and Web service of HydroMP).

4.1 Design of Hydro1D data structure

HydroMP decomposes the complex systems for abstract description and hierarchical storage based on an object-oriented modeling method (Booch et al., 2007). Two association structures, i.e. single scenario storage and multi-scenario storage based on the object-oriented program, were proposed. The multi-scenario storage aims at reducing the amount of data transmissions during the submission of multiple scenarios. In general, the different scenarios have the same base data including the river network and topology, and the differences between different scenarios are the upper river segment discharges and boundary conditions. In multi-scenario storage, each project includes one network structure and multiple scenario groups designed on this river network structure. Each scenario group includes a set of scenarios of a specific type. In this way, a large amount of traffic data can be saved in the submission of multiple scenarios via the Web, and the topology of the data structure is shown in Figure 9. In a single scenario storage structure, one scenario stores the single river network structure, the single cross section scenario, the single boundary scenario, the single building scenario, the single control scenario and simulation parameter scenario in parallel. The data structure of objects in the second layer is the same as that in a multi-scenario storage.

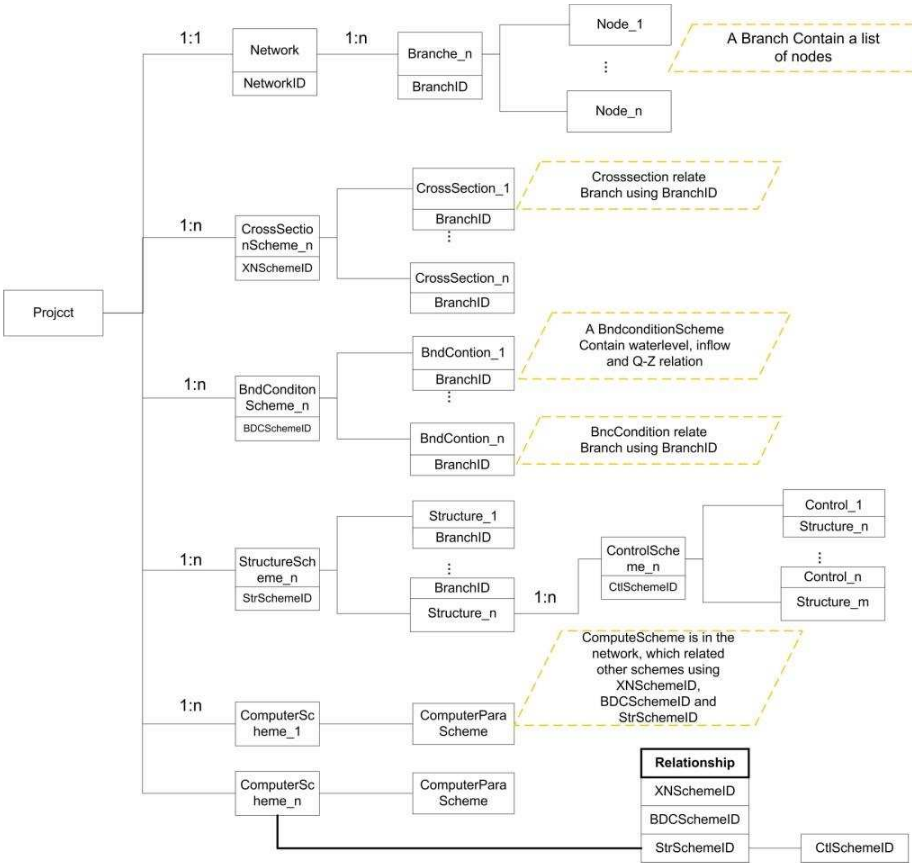


Fig. 9. Data structure of the one-dimensional hydrodynamic model

4.2 Registered hydro-models and model management

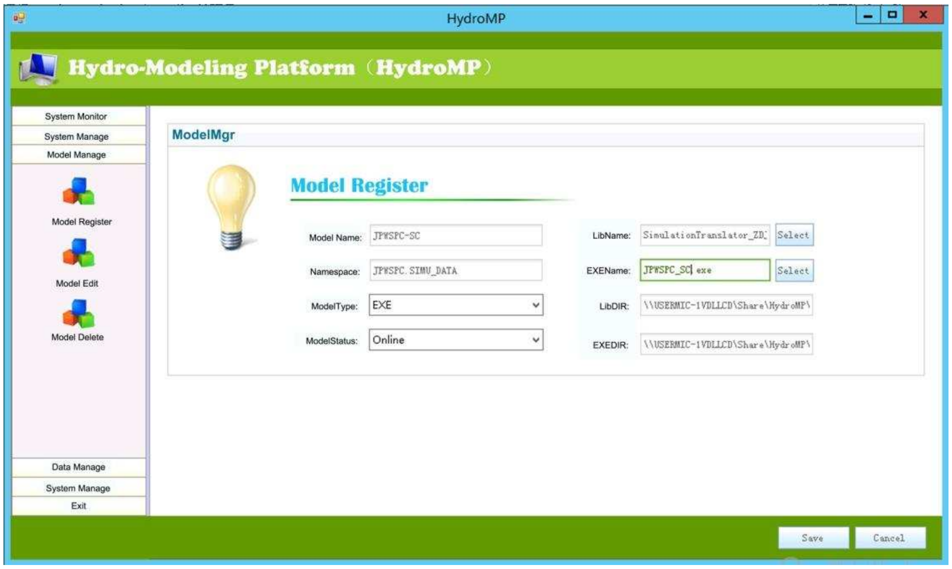
The platform implements the integration of CE-QUAL-RIV1[‡] and JPWSPC (Zhu et al., 2011) through the standardized interfaces. The CE-QUAL-RIV1 model is a one-dimensional hydrodynamic model developed by the U.S. Army Corps, and in the present study it is integrated using the EXE mode. The JPWSPC model is a river network hydrodynamic model proposed by Zhu. This model uses a junction point water stage prediction-correct method to solve the complex river network modeling (Zhu et al., 2011). In the present study, the integration of JPWSPC is achieved separately using SPIIM and PPIIM.

CE-UQAL-RIV1 is integrated in an EXE mode (non-invasive). The InputFileCreate(), GetProcess(), and OutFileRead() interface was implemented, and the GetProcess() interface is implemented based on the relations between the output file size and simulation process. The JPWSPC model was rewritten using PIIM. The original program was rewritten into four functions, Initialize(), PerformTimeStep(), UpdateBndData() and GetTimeStepResult().

In addition, in the present study, the parallel integration of the JPWSPC model was also implemented. In parallel programs, MPI is used for communication between the master and slave processes, and the method and interface for communication between the master process and the platform (computing scheduler) are the same as in the PIIM mode.

The system uses a database to keep information on the models that have already been integrated, including the model name, the model type, the location of execution files and the data exchange DLL files. The hydro-models can be registered and cancelled dynamically by changing the hydro-model state in the database. and Figure 10 shows the model registration GUI.

[‡] http://www.epa.gov/athens/wwqtsc/html/epd-riv1.html.



Model Name: JPWSPC-SC
Namespace: JPWSPC.SIMU_DATA
ModelType: EXE
ModelStatus: Online
LibName: SimulationTranslator_ZD
EXEName: JPWSPC_SC exe
LibDIR: \\USERMIC-1VDLLCD\Share\HydroMP\
EXEDIR: \\USERMIC-1VDLLCD\Share\HydroMP\

Fig. 10. Graphic user interface for model registration

To test the concurrent capacity and Web service of the platform, a desktop client based on C#, CloudHydro1D, was developed, which provides the functions for editing the multi-scenario data, submitting the scenario simulations and simulation results display.

5 CASE STUDY

HydroMP is designed for the multi-user, multi-scenario concurrent simulations. To test the platform's capacity for rapid calculation and concurrency management, the simulation of real-time scheduling optimization in the SNWD (South-North Water Diversion) project was used as an example case. The concurrent management of multiple scenarios and the relations of different concurrencies with calculation times and response times were examined.

5.1 Project introduction

SNWD is a major water allocation project to solve the water shortages in northern China. The central route diverts water from the Taocha hub of the Danjiangkou Reservoir, and runs to

Beijing with a total length of 1277 km. Along all the routes, there are 61 sluices, 78 outlets, and 42 back outlets. The project diverts water in an open channel; during water diversion, changes in the water flow at any outlet or in the opening of any sluice will lead to changes in the hydrodynamic process downstream or even along the entire route. A schematic view of the route of SNWD is shown in Figure 11. To ensure the safety of water diversion, the water flow adjustment and control strategy of the main route via the opening and closing of sluices, outlet gates and back outlet gates should be applied in the channel operation to maintain a relative constant upstream water level of sluices and to reduce the complex hydraulic response in the negative feedback way. The method of controlling the opening and closing of the sluices, outlet gates and back outlet gates (timing of opening and closing, and gate opening angle) is a complex problem, and an optimal or suboptimal adjustment scenario can only be obtained through testing and optimizing different gate opening and closing scenarios using hydrodynamic models and gate control models. In particular, when the flows at certain water-drawing stations change drastically, the question of how to control the gates along the route to ensure the fastest recovery to a water drawing-water pumping balance is an issue that must be solved during the real-time scheduling process. The real-time sluice operation at different levels and relevant decision-making processes in SNWD require a most optimum solution plan among enormous viable options in a multi-dimensional scenario. It demands extensive computing resources to satisfy the promptness and efficiency of various regulation procedures, which would be impossible to implement under conventional serial computing environment. The current level of water diversion service along the middle route is achieved through the standard practice without multi-scenario simulation and multi-system analysis and the operational mode is in series but not in parallel. Thus it is difficult to realize the optimum regularisation and efficient management control. The HydroMP system will be able to improve the situation by overcoming these existing limitations. Thus here we will explore the potentials of cloud computing based on HydroMP platform in this study to demonstrate the applicability of parallel computing technique and its efficiency in engineering practice. In the present study a multi-scenario automatic comparison method was used for hydrodynamic simulation, verification and optimization of multiple gate control scenarios in order to obtain the optimized scenario.

5.2 Scenarios description

Due to the limitations set by the amount of water available at water-pumping stations and the varying amounts of water needed in the water importing area, water diversion in the main canal is an uneven process. Changes in the water flow at any canal segment will cause fluctuations in the canal's water level. In the present study, hydraulic transition processes under certain extreme operating modes were investigated: the water flow at the Taocha headworks linearly changed to another level within a certain amount of time; the water flows at different outlets changed linearly and synchronously; the different sluices were opened from the initial opening angle to the target opening angle.

Fig. 11. The middle route of South-North Water Diversion project

Flow at the Taocha headworks has two operating modes: decrease and increase: (1) Assume that in the initial state the canal contains 70% of the design flow, and in a given amount of time it is reduced to 50% and 10% of the flow; (2) Assume that in the initial state the canal contains 70% and 10% of the design flow, which respectively increases to 80% and 70% of the design flow. Based on the amplitude and time of flow changes at the headworks, four operating modes are planned. For operating modes 1 and 2, the gate opening needs to be reduced, whereas for operating modes 3 and 4, the gate opening needs to be enlarged. Detailed parameters of the operating modes are shown in Table 2.

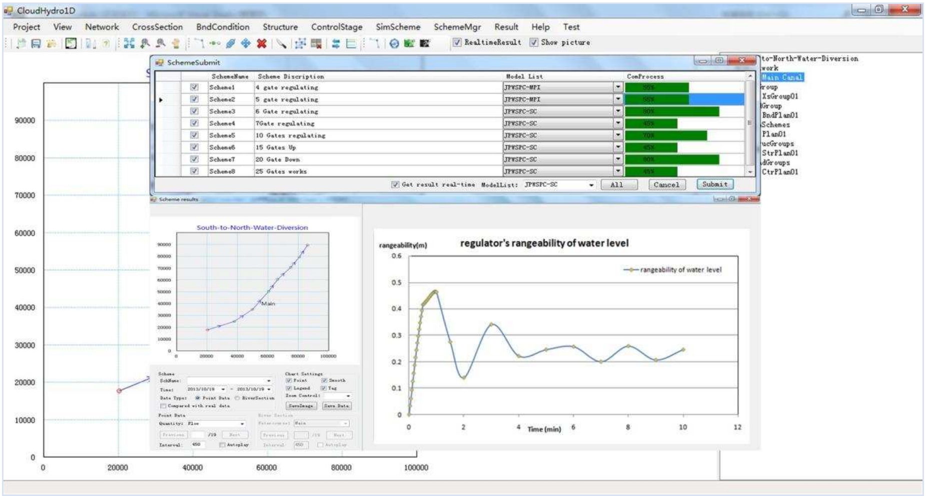
Table 2. Parameters for water flow changes in the main canal under different operating modes

Operating mode	flow change at headworks		Time of flow changes at headworks (min)	Time of gate opening changes (min)	Time of flow changes at outlets (min)	Scheduling strategy	Computing model
	starting time	ending time					
1	70%	50%	10	10	10	Sequential control	JPWSPC-PC
						Synchronized control	JPWSPC-PC
						Temporal sequence control	JPWSPC-PC
2	70%	10%	30	30	30	Sequential control	JPWSPC-MPI
						Synchronized control	JPWSPC- MPI
						Temporal sequence control	JPWSPC- MPI
3	70%	80%	10	10	10	Sequential control	JPWSPC-PC
						Synchronized control	JPWSPC-PC
						Temporal sequence control	JPWSPC-PC
4	10%	70%	30	30	30	Sequential control	JPWSPC- MPI
						Synchronized control	JPWSPC- MPI
						Temporal sequence control	JPWSPC- MPI

In the above conditions, different scheduling strategies and computing models were used. The total combinations included 12 scenarios, consisting of 12 different processes for gate opening/closing. Through the Web service interface provided by the HydroMP platform, the scenarios were submitted to the cloud end for calculation. Based on the feedbacks from the calculation results, upstream water levels and the maximum amplitude of changes in gate opening were evaluated. Gate opening/closing was fine-tuned accordingly before beginning the next round of calculations. The number of adjustment was set to 10, and the optimal scenario was selected among the last fine-tuned group of scenarios.

5.3 Multi-scenario submission

In addition to the backstage submissions, the PC terminal CloudHydro1D provides a user interface for data editing, scenario submissions and result displays. After entering the "scenario submission" GUI, the system lists, in real-time, all scenarios in the project and acquires the model name that can be used for calculations according to the scenario type from the model library in the HydroMP server. As the current scenario is a one-dimensional canal hydrodynamic scenario, models that can be selected by the HydroMP server include CE-QUAL-RIV, JPWSPC-SC and JPWSPC-MPI. During the scenario computing, CloudHydro1D calls the GetProcess() interface for real-time access to computing progress and results, as shown in Figure 12. In Figure 12, the interface shows the time series of the average amplitude of upstream water level at one outlet.



	SchemeName	Scheme Discription	Model List	ComProcess
☑	Scheme1	4 gate regulating	JPWSPC-MPI	55%
☑	Scheme2	5 gate regulating	JPWSPC-MPI	55%
☑	Scheme3	6 Gate regulating	JPWSPC-SC	80%
☑	Scheme4	7Gate regulating	JPWSPC-SC	45%
☑	Scheme5	10 Gates regulating	JPWSPC-SC	70%
☑	Scheme6	15 Gates Up	JPWSPC-SC	45%
☑	Scheme7	20 Gate Down	JPWSPC-SC	80%
☑	Scheme8	25 Gates works	JPWSPC-SC	45%

☑ Get result real-time ModelList: JPWSPC-SC All Cancel Submit

Fig. 12. Interface for scenario submission

5.4 Scenario results

In addition to a real-time display of calculation results during the computing, CloudHydro1D also provides tools for the result demonstrations and analysis. The result demonstration tool can be used to display the time series of flow and water level at a single point, water level animation at a single cross section, branch flow profile and animation. The scenario analysis tool features a function allowing the user to compare the results of different models and different scenarios. The result analysis tool is a small tool for result evaluation and the user can first set specific measured data at certain points, and then observe the automatic matching between the measured data and the simulation results at these monitoring points through coordinate information.

5.5 Test on the efficiency of HydroMP for concurrent simulation

To test the capacity of the platform for concurrent processing, four different concurrent scenarios were set: 3, 5, 10 and 20 users logged in simultaneously, respectively corresponding to 150, 500, 800 and 1000 concurrent scenarios submitting, and 800, 1200, 1500 and 1600 initialized processes. The test results showed that the average response time was no longer than 0.4s, and the average computing time of each single scenario was 4.5s, 4.8s, 5.2s and 5.32s, as illustrated in Table 3. These results suggest that with the current hardware architecture and scheduling methods, the platform has the capacity to process several hundred concurrent scenarios, yet the response time increases notably after the number of concurrent scenarios submitted exceeds 500, indicating there is competition for resources in the server where the HydroMP center management service system is located. In the future, concurrency needs to be tested further, especially on the processing capacity and hardware requirement of the HydroMP center after expansion of HydroMP server.

Table 3. Response times in different concurrent submission scenarios

No.	**Number of users**	**Number of concurrent scenarios**	**Number of initialized processes**	**Average submission response time (s)**	**Average computing time (s)**
1	3	150	800	0.08	4.5
2	5	500	1200	0.25	4.8
3	10	800	1500	0.35	5.2
4	20	1000	1600	0.38	5.32

6 CONCLUSION AND DISCUSSION

Over the last three years, some researchers have recognized the promoting effect of cloud computing on water management and simulation computing, and have begun to develop a number of model integration and simulation service platforms based on the cloud computing and Web services. These platforms can be divided into three categories. One is a framework for data sharing and collaborative decision-making, e.g. HydroDesktop. The second type provides cloud service with a simulation function. These platforms use HPC Clusters similar to the one used in the present study for multi-use, multi-scenario concurrent computing. The third category is a platform based on model sharing and model coupling, e.g., the CSDMS system.

The cloud computing service mode proposed in the present paper shares certain commonalities with the above modes, e.g., use of HPC Cluster as a computing resource, and use of a Web service as a service interface. However, the method proposed in the present paper also has unique new characteristics, including the following: (1) it focuses on the dynamic integration of data exchange between the platform and the model, thereby providing the "single process, selection of multiple models" mode to the model users. This gives more choices to the users that need large-scale simulations compared with the other systems. The users can select the most appropriate model according to the scope, the computing times and other attributes of the different integrated models; (2) Regarding to the universal model integration, the HydroMP platform provides both a coarse-grained EXE integration approach and a program interactive integration method, using a standard component wrapper to implement the integration of model components as well as standardized communication between the model and the platform. Thus, the platform not only adapts to the future trend of multi-disciplinary model coupling and integration, but also enables the non-rewriting integration of legacy models, thereby providing more choices for the model developers than other systems. (3) The HydroMP platform proposed in the present paper can be subjected to distributed deployment; through a HydroMP center, the load balancing between the various HydroMP servers is achieved, thereby ensuring computing resource scalability; (4) According to the time series model characteristics of hydraulic simulation in the HydroMP system, pause, restart and other methods are used to assist computing resources scheduling to meet the emergency scenario simulation. Using this method, the amount of computation required does not increase and the priority of emergency scenario is ensured.

Real-time optimization simulations of the SNWD and test results on platform performance show the robust model integration method, computing resources scheduling rule and the processing capacity of concurrent simulation. However, it should be noted that HydroMP needs improvement in security mechanisms and integration with other data and computing service platforms. Specifically, (1) A traditional mechanism, i.e., the password checking mechanism, is still used for the safe communication between the platform and the terminal, but cannot meet the safety requirements for data storage and transmission; (2) The integration between the HydroMP platform and other service platforms (i.e. HydroServer and other data sharing platforms) has not been realized; (3) The response speed for scenario simulation in HydroMP also depends on the network bandwidth. In the current testing of case study, the terminals and the HydroMP platform are in the same LAN, and the communication speed between the cloud and the terminal is 100 Mb/s in theory.

The efficiency of parallelization heavily depends on the ratio between the computing load and the data access and communication. The larger the ratio is, the higher the parallelization efficiency could become. Compared with the 1D modelling, the ratios in 2D &3D are much larger and thus we believe the latter should achieve higher parallelization benefit and thus have more potential for the proposed HydroMP platform.

Currently our HydroMP system is not an open accessed one. However, a trial version is being commercialised and the relevant website is being constructed to collect user feedbacks aiming to further improve the HydroMP platform service.

Acknowledgements

This work has been sponsored in part by the National Natural Science Foundation of China (51459003, 51579131), National Key Technology Research and Development Program of the Ministry of Science and Technology of China (Grant No. 2013BAB05B03, 2013BAB05B05), Chinese Ministry of Water Resources special funds for scientific research on public causes (Grant No. 201201050 and No. 201501028), the State Grid Qinghai Electric Power Company (Grant No. SGTYHT/14-JS-188), and IWHR Research & Development Support Program (JZ0145B042016). We would like to thank all our sponsors and other members of the HydroMP and Cloud Computing Research Group at the State Key Laboratory of Hydroscience and Engineering, Tsinghua University.

REFERENCES

Ames, D.P., Horsburgh, J.S., Cao, Y., Kadlec, J., Whiteaker, T., Valentine, D., 2012. HydroDesktop: Web services-based software for hydrologic data discovery, download, visualization, and analysis. Environmental Modelling& Software 37, 146-156.

Arango, I.M., Izquierdo, J.S., Campbell, E.O.G., Pérez-García, R., 2014. Cloud-based decision making in water distribution systems. Procedia Engineering 89, 488-494.

Ari, I., Muhtaroglu, N., 2013. Design and implementation of a cloud computing service for finite element analysis. Advances in Engineering Software 60-61, 122-135.

Blöschl, G., Reszler, C., Komma, J., 2008. A spatially distributed flash flood forecasting model. Environmental Modelling & Software 23(4), 464-478.

Booch, G., Maksimchuk, R.A., Engle, M.W., Young, B.J., Conallen, J., Houston, K.A., 2007. Object-Oriented Analysis and Design with Applications, Third Edition. Addison Wesley.

Brooking, C., Hunter, J. 2013.Providing online access to hydrological model simulations through interactive geospatial animations. Environmental Modelling & Software 43, 163-168.

Bulatewicz, T., Allen, A., Peterson, J.M., Staggenborg, S., Welch, S.M., Steward, D.R., 2013. The Simple Script Wrapper for OpenMI: Enabling interdisciplinary modelling studies. Environmental Modelling & Software 39, 283-294.

Bürger, C.M., Kollet, S., Schumacher, J., Böselc, D., 2012. Introduction of a web service for cloud computing with the integrated hydrologic simulation platform ParFlow. Computers & Geosciences 48, 334-336.

Caballer, M., Alfonso, C., Alvarruiz, F., Moltó, G., 2013. EC3: Elastic Cloud Computing Cluster. Journal of Computer and System Sciences 79(8), 1341-1351.

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.

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.

David, O., Ascough II, J.C., Lloyd, W., Greenc, T.R., Rojasd, K.W., Leavesleya, G.H., Ahujac, L.R., 2013. A software engineering perspective on environmental modeling framework design: The Object Modeling System. Environmental Modelling & Software 39, 201-213.

Glenis, V., McGough, A.S., Kutija, V., Kilsby, C., Woodman, S., 2013. Flood modelling for cities using Cloud computing. Journal of Cloud Computing: Advances, Systems and Applications 2(1): 7.

Gregersen, J.B., Gijsbers, P.J.A., Westen, S.J.P., 2007. OpenMI: open modelling interface. Journal of Hydroinformatics 9(3), 175-191.

Gupta, P., Seetharaman, A., Rudolph Raj, J., 2013. The usage and adoption of cloud computing by small and medium businesses. International Journal of Information Management 33(5), 861-874.

Huang, Q., Yang, C., Liu, K., Xia, J., Xu, C., Li, J., Gui, Z., Sun, M., Li, Z., 2013. Evaluating open-source cloud computing solutions for geosciences. Computers & Geosciences 59, 41-52.

Hwang, K., Dongarra, J., Fox, G.C., 2011. Distributed and Cloud Computing: From Parallel Processing to the Internet of Things. Morgan Kaufmann, San Francisco.

Laniak, G.F., Olchin, G., Goodall, J., Voinov, A., Hill,M., Glynn, P., Whelan,G., Geller,G., Quinn,N., Blind, M., Peckham,S., Reaney,S., Gaber, N., Kennedy, R., Hughes, A., 2013. Integrated environmental modeling: A vision and roadmap for the future. Environmental Modelling & Software 39, 3-23.

Launder, B.E., Spalding, D.B., 1974. The numerical computation of turbulent flows. Computer Methods in Applied Mechanics and Engineering 3(2), 269-289.

Lloyd, W., David, O., Lyon, J., Rojas, K.W., Ascough II, J.C., Green, T.R., Carlson, J.R., 2012. The Cloud Services Innovation Platform-Enabling Service-Based Environmental Modelling Using Infrastructure-as-a-Service Cloud Computing. In 2012 International Congress on Environmental Modelling and Software, Managing Resources of a Limited Planet, Sixth Biennial Meeting, Leipzig, Germany.

Overeem, I., Berlin, M.M., Syvitski, J.P.M., 2013. Strategies for integrated modeling: The community surface dynamics modeling system example. Environmental Modelling & Software 39, 314-321.

Reed, S., Schaake, J., Zhang, Z., 2007. A distributed hydrologic model and threshold frequency-based method for flash flood forecasting at ungauged locations. Journal of Hydrology 337(3-4), 402-420.

Rodríguez, Á., Llort, X., Sancho, D., Sánchez-Diezma, R., Bella, R., Gómez, V., 2014. HIDROMET: A Cloud-based EWS platform for real time urban flood warning. In 11th International Conference on Hydroinformatics, HIC 2014, New York, USA.

Schmitz, O., Salvadore, E., Poelmans, L., van der Kwast, J., Karssenberg, D., 2015. A framework to resolve spatio-temporal misalignment in component-based modelling, Journal of Hydroinformatics, 16(4), 850-871.

Shi, H.Y., Li, T.J. Liu, R.H., Chen, J., Li, J.Y., Zhang, A., Wang, G.Q., 2015. A service-oriented architecture for ensemble flood forecast from numerical weather prediction. Journal of Hydrology 527, 933-942.

Yizi Shang, Y.Z., Guo, Y.X., Shang, Li., Ye, Y.T., Liu, R.H., Wang, G.Q., 2016. Processing conversion and parallel control platform: a parallel approach to serial hydrodynamic simulators for complex hydrodynamic simulations. Journal of Hydroinformatics 18(5),851-866.

Silva, P.H.L., Freitas, G. A.A., Carvalho-Junior, F.H., Corrêa, R.C., 2012. Connection Mechanisms of CCA Components for Parallel Applications, In: 13th Symposium on Computer Systems, Petropolis, Brazil. pp. 210-217.

Sun, A., 2013. Enabling collaborative decision-making in watershed management using Cloud-computing services. Environmental Modelling & Software 41, 93-97.

Wang, B., Fringer, O.B., Giddings, S.N., Fong, D.A., 2009. High-resolution simulations of a macrotidal estuary using SUNTANS. Ocean Modelling 26(1-2), 60-85.

Wei, C.C., Hsu, N.S., 2008. Multi reservoir real-time operations for flood control using balanced water level index method. Journal of Environmental Management 88(4), 1624-1639.

Wu, Z.C., Wang, D.Z., 2010. Simulation of the oil slick movement in tidal water-ways. Journal of Hydrodynamics, Ser. B 22(1), 96-102.

Zeng, W., Song, Q., Liu, H., Wang, T., 2010. Research on ANN-based Pre-warning Water Bloom Model of LiuHai Lake in Beijing. In: International Conference on Ecological Informatics and Ecosystem Conservation (ISEIS 2010), Procedia Environmental Sciences 2, pp.625–635.

Zhu, D., Chen, Y., Wang, Z., Liu, Z., 2011. Simple, Robust, and Efficient Algorithm for Gradually Varied Subcritical Flow Simulation in General Channel Networks. Journal of Hydraulic Engineering (ASCE) 137(7), 766-774.