water

Article

Development of a Prototype Web-Based Decision Support System for Watershed Management

Dejian Zhang [1], Xingwei Chen [1,2,*] and Huaxia Yao [3]

[1] College of Geographic Sciences, Fujian Normal University, Fuzhou 350007, China;
E-Mail: djzhang80@163.com

[2] Cultivation Base of State Key Laboratory of Humid Subtropical Mountain Ecology,
Fujian Normal University, Fuzhou 350007, China

[3] Dorset Environmental Science Centre, Ontario Ministry of Environment and Climate Change,
1026 Bellwood Acres Road, Dorset, ON P0A 1E0, Canada; E-Mail: Huaxia.Yao@ontario.ca

* Author to whom correspondence should be addressed; E-Mail: cxwchen215@163.com;
Tel.: +86-591-8353-9685; Fax: +86-591-8346-5397.

Abstract: Using distributed hydrological models to evaluate the effectiveness of reducing non-point source pollution by applying best management practices (BMPs) is an important support to decision making for watershed management. However, complex interfaces and time-consuming simulations of the models have largely hindered the applications of these models. We designed and developed a prototype web-based decision support system for watershed management (DSS-WMRJ), which is user friendly and supports quasi-real-time decision making. DSS-WMRJ is based on integrating an open-source Web-based Geographical Information Systems (Web GIS) tool (Geoserver), a modeling component (SWAT, Soil and Water Assessment Tool), a cloud computing platform (Hadoop) and other open source components and libraries. In addition, a private cloud is used in an innovative manner to parallelize model simulations, which are time consuming and computationally costly. Then, the prototype DSS-WMRJ was tested with a case study. Successful implementation and testing of the prototype DSS-WMRJ lay a good foundation to develop DSS-WMRJ into a fully-fledged tool for watershed management. DSS-WMRJ can be easily customized for use in other watersheds and is valuable for constructing other environmental decision support systems, because of its performance, flexibility, scalability and economy.

1. Introduction

Climate change, population growth and unreasonable exploitation of water resources have caused environmental deterioration, the unavailability of freshwater and an imbalance between supply and demand to a global extent, thus seriously affecting the sustainable development and utilization of water resources. At present, more than 1.2 billion people and 60% of global basins lie at the edge of water resource shortage [1]. How to relieve or eliminate the deterioration of the water environment and realize the sustainable utilization of water resource have become common concerns of and challenges for humankind. Scientific and effective tools are urgently needed to fulfill the purpose of the sustainable utilization of water resource. The decision support system for watershed management (DSS-WM) is one of the representative management tools and plays an important role in watershed management.

Driven by the latest advancements of information and communication technologies, hydrologic sciences and other disciplines, there is booming research on watershed management using hydrological models. For example, distributed hydrological and hydrodynamic models, such as SWAT (Soil and Water Assessment Tool), HSPF (Hydrological Simulation Program Fortran), AGNPS (agricultural non-point source pollution model) and WASP (Water Quality Analysis Simulation Program), are all geared with management modules for the simulation and evaluation of management effects on flow, sediment and nutrients [2–6]. Although great achievements have been gained by these models, the complex model structures and interfaces have impeded their applications by inexperienced users. Besides, the time-consuming and computationally costly procedures of model simulations have further hindered the application of these models, especially under circumstances where real-time or quasi-real-time support for decision-making is required.

To overcome the aforementioned shortcomings, hydrologists and environmental scientists have designed and invented many dedicated DSS-WMs to assist with watershed management. Similar to other environmental DSS, these DSS-WMs usually consist of a decision-making information database and user interfaces and models [7–9]. According to the operational environments, DSS-WMs can be divided into desktop-based and web-based. Desktop-based DSS-WMs usually provide intuitive wizard style interfaces, which eliminate the complexity of the models. For example, under the impetus of the MULINO (Multi-sectoral, Integrated and Operational DSS) project, Mysiak *et al.* [10] developed mDSS (a decision support system for water resource management that has been developed under the European research project, MULINO) for optimizing the management of water resources by integrating hydrological models with multiple-criteria evaluation procedures. Cau and Paniconi [11] linked SWAT and mDSS to assess four alternatives, including intensive agriculture and dairy farming and treated wastewater for irrigation. Hipel *et al.* [12] designed and developed the GMCR II (graph model for conflict resolution) for conflict resolution over multiple stakeholders in controlling water pollutions.

It is a general trend to turn to the Internet as a platform for software solutions, and so is DSS-MWs. Rao *et al.* [13] developed a prototype web-based DSS based on a commercial Web GIS (Web-based Geographical Information Systems) tool, ArcIMS (Arc Internet Map Server), and a hydrological model, SWAT. Additionally, the prototype was then applied to a small watershed, Panhandle in Oklahoma, targeted at aiding a better management plan. Model parallel simulation and a cloud computing platform were not attempted in their work. Zeng *et al.* [14] constructed a web-based decision-making system by integrating the ArcGIS Engine, the distributed hydrological model, Hydrologic Engineering Center's Hydrologic Modeling System (HEC-HMS), genetic algorithm (GA) and artificial neural network (ANN). HEC-HMS was applied to the prediction of runoff; ANN was used to predict the city water resources demand; GA was used to achieve the goal of distributing water resources among the regions of the city. Sun [15] migrated a web-based DSS to a public cloud, which is an extension of web-based solutions.

Throughout the development of DSSs for watershed management, great achievements have been made by integrating models and other technologies for better watershed management. However, there are still some inadequacies, such as: (1) Most integrated models are conceptual or empirical models, and distributed hydrological ones are few; (2) More DSSs for watershed management are desktop-based, while the web-based ones are still rare; and (3) The performance of the systems are not well explained or evaluated, which is a key factor to achieve the goal of real-time decision support.

Our objectives in this study are to design and develop a web-based decision support system for watershed management (DSS-WMRJ), which is user friendly and supports quasi-real-time decision making. We build the DSS-WMRJ by integrating an open source Web GIS tool (Geoserver), a modeling component (SWAT, Soil and Water Assessment Tool), a cloud computing platform (Hadoop) and other open source components and libraries. In addition, a private cloud is used in an innovative manner to parallelize the model simulations, which are time consuming and computationally costly. The successful implementation and testing of the prototype DSS-WMRJ shows that it is able to fulfill the goal of quasi-real-time decision support and provide intuitive interfaces.

2. System Design

2.1. Architecture of Decision Support System for Watershed Management (DSS-WMRJ)

To meet the requirements of availability, stability, interoperability and portability, a systematic architecture of four tiers, including the presentation, proxy, application and database and model, is considered (Figure 1).

The presentation tier provides a graphic user interface, which is accessible via the browsers of many devices, for users to perform system management, map operations, spatial and attribute information retrieval, watershed management, and so on. The map viewer is achieved by the Openlayers component, which communicates with map services to retrieve the grid or vector map through Asynchronous JavaScript and XML (AJAX) and to render the map in the browser. Thus, it provides operation experience approximate to a desktop GIS tool. FusionCharts is the only commercial software used for presenting the watershed management results, due to its dynamic and excellent chart functionalities.
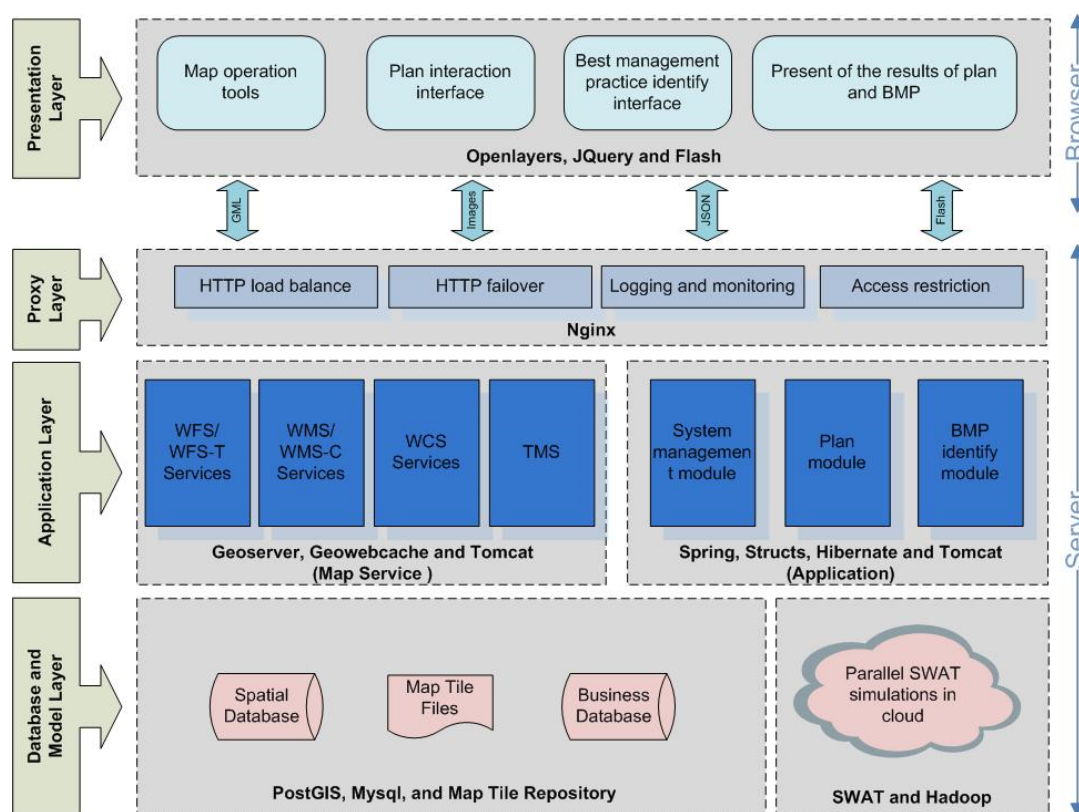
Figure 1. The system architecture of decision support system for watershed management (DSS-WMRJ). WFS/WFS-T, Web Feature Service/Web Feature Service-Transaction; WMS, Web Map Service; WCS, Web Coverage Service.

The open-source component Nginx is used as the proxy tier, which lies between the presentation and application tiers and acts as a communication agent for these two tiers. The deployment and configuration of Nginx is easy, while it provides useful functionalities, such as load balancing, failover, access control, logging, monitoring, *etc*. When the system exceeds the workload limit of the system, the system administrator can add more background services and a simple configuration of Nginx to scale up the system. Therefore, the proxy tier is very important for enhancing the performance and improving the stability of the system.

The application tier consists of two components: map service and watershed management services. Both components adopt the Service-Oriented Architecture (SOA). The map services component uses an open source Web GIS tool (Geoserver), which is in compliance with the Open Geospatial Consortium (OGC) standards, such as Web Map Service (WMS), Web Feature Service/Web Feature Service-Transaction (WFS/WFS-T) and Web Coverage Service (WCS). The watershed management services provide functionalities, such as planning and BMP identification. These components are standard compliant and service oriented, making them scalable and interoperable.

The database and model tier is located on the bottom of the architecture. This tier consists of databases and model simulation services. The databases store and manage attribute data, spatial data and map tiles via a spatial database, an object-relational database and a file system. The spatial data are stored in the PostGreSQL database with the use of the PostGIS library, which adds support for the use and management of geographic objects. Spatial and other regular indices are created for every map layer

stored in the spatial database to increase the speed of retrieval. Map tiles are pre-generated and stored in the map tile repository. This will accelerate the mapping processes, as WMS can directly deliver the caching map tiles to the client when a map request is sent to it. In addition, the model simulation service is a key component of the DSS-WMRJ, which guarantees quasi-real-time decision making by parallelizing model simulations on a private cloud. A detailed description of the model simulation service is given in the next section.

2.2. Model Simulation Service

The decision making procedures usually require a great many model simulations, for example, when an uncertainty analysis is required in the decision making as the model input, the structure and parameters contain various degree of uncertainty or when evaluating the environmental effect of combinations of different management measures, which may themselves involve different configurations (making the decision making procedures very time consuming and computationally costly). Therefore, a fast model simulation is the key factor of DSS-WMRJ to achieve real-time or quasi-real-time decision making.

An open-source cloud computing platform (Hadoop) is used to parallelize model simulations in order to accelerate the simulation procedures. Hadoop is an implementation of the Google MapReduce algorithm [16,17]. It consists of two components: the Hadoop Distributed File System (HDFS) and the distributed computation framework (MapReduce). HDFS is a robust distributed file system, which is able to read and write data in parallel over a large number of machines and achieves much higher throughput than traditional technologies. This feature is very useful to process mass model simulation results. MapReduce is a distributed computing framework that consists of the JobTracker and the TaskTrackers. It provides two important application programming interfaces (APIs): Mapper and Reducer. With these interfaces, developers can quickly write efficient parallel codes. Hadoop parallelizes tasks as follows: (1) Clients submit a job to the JobTracker, which is the master of the MapReduce framework; (2) The JobTracker then divides the submitted job into task sets and distributes these task sets to TaskTrackers; and (3) the tasks in the assigned set are further distributed to Mapper or Reducer, which then executes the task.

To achieve SWAT parallel simulations on a Hadoop cluster, developers must implement the aforementioned APIs of MapReduce, and co-operation among the presentation tier, application service and model simulation service is needed. Figure 2 shows the procedures of paralleling SWAT simulations. These procedures are summarized as follows: (1) A user is prompted for certain specific inputs that pertain to management practices and submits these inputs to the application service; (2) The application service translates the inputs into parameter sets of the SWAT model and distributes these parameter sets to the model simulation service; (3) The model simulation service parallelizes the model simulations, which involves operations, such as model input file editing, model executing, simulation result extracting and saving results to the HDFS; (4) When the submitted job is finished, the application service gathers all simulation results in the HDFS and generates a statistic report, which is XML-based, and delivers it to the presentation tier; and (5) Finally, the presentation tier renders the report through its chart component.
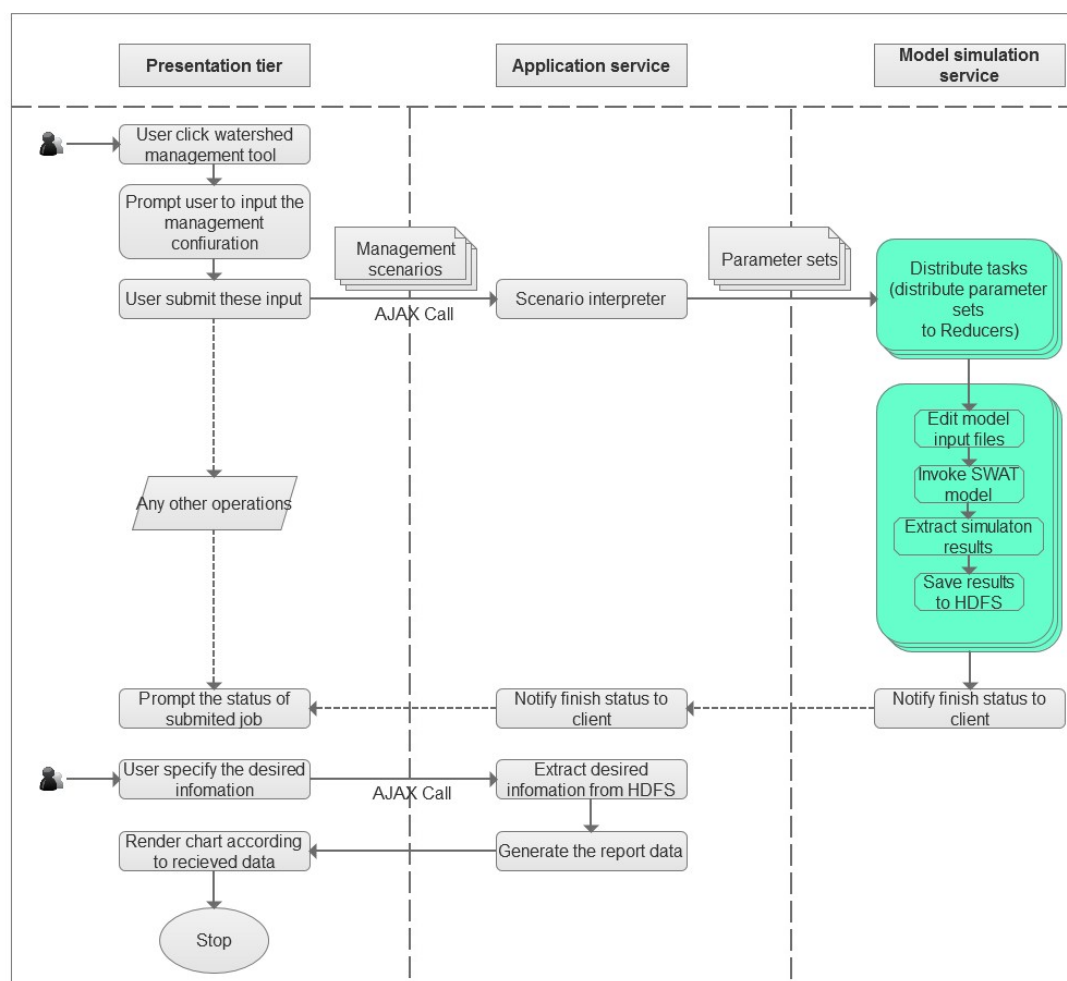
Figure 2. Parallelizing the simulation of management scenarios (the colored parts are the execution in parallelizing). AJAX, Asynchronous JavaScript and XML; HDFS, Hadoop Distributed File System; SWAT, Soil and Water Assessment Tool.

3. Case Study and DSS-WMRJ Test

3.1. Model Setup

The SWAT model [18,19] is a semi-distributed, continuous, watershed-scale hydrological model that was developed by the Agricultural Research Service of the United States Department of Agriculture (USDA-ARS) to simulate the quantity and quality of surface water and groundwater. It not only deeply depicts the physical hydrological cycle, but also considers the impact of human activities, such as land use change, water conservancy facilities, agriculture management practices and other environment protecting facilities (e.g., vegetation filter strips and grassed waterways) on the hydrological processes.

Jinjiang basin with an area of 5629 km^2 is selected as the test watershed for which to implement and evaluate the DSS-WMRJ. ArcSWAT, one of the graphical user interface procedures for SWAT, is used to delineate Jinjiang basin [20]. The basin is divided into 99 subbasins based on the DEM data and with a threshold area of 3000 ha. The subbasins are subdivided into HRUs, which represent homogeneous soil and land use according to the soil type, land use and topographic slope, with threshold values of 5%, 20% and 20%, respectively, resulting in 886 HRUs. Additionally, the watershed model is set to run in

daily mode. The SWAT has been calibrated based on water discharge data, but not calibrated for sediment and nutrients, because of the insufficient monitoring data.

3.2. System Implementation

According to the design scheme of DSS-WMRJ, a prototype of DSS-WMRJ was established by incorporating the hydrological model of an experimental watershed. Figure 3 is the GUI of the prototype of DSS-WMRJ. The left column provides the functionalities of the system and layer management. The system functionalities control the privileges of users, and the layer management controls the switching on or off of layers. The right column provides watershed management functionalities and some general map-related functionalities, such as map roaming, zoom in/out, overview map, and so on.
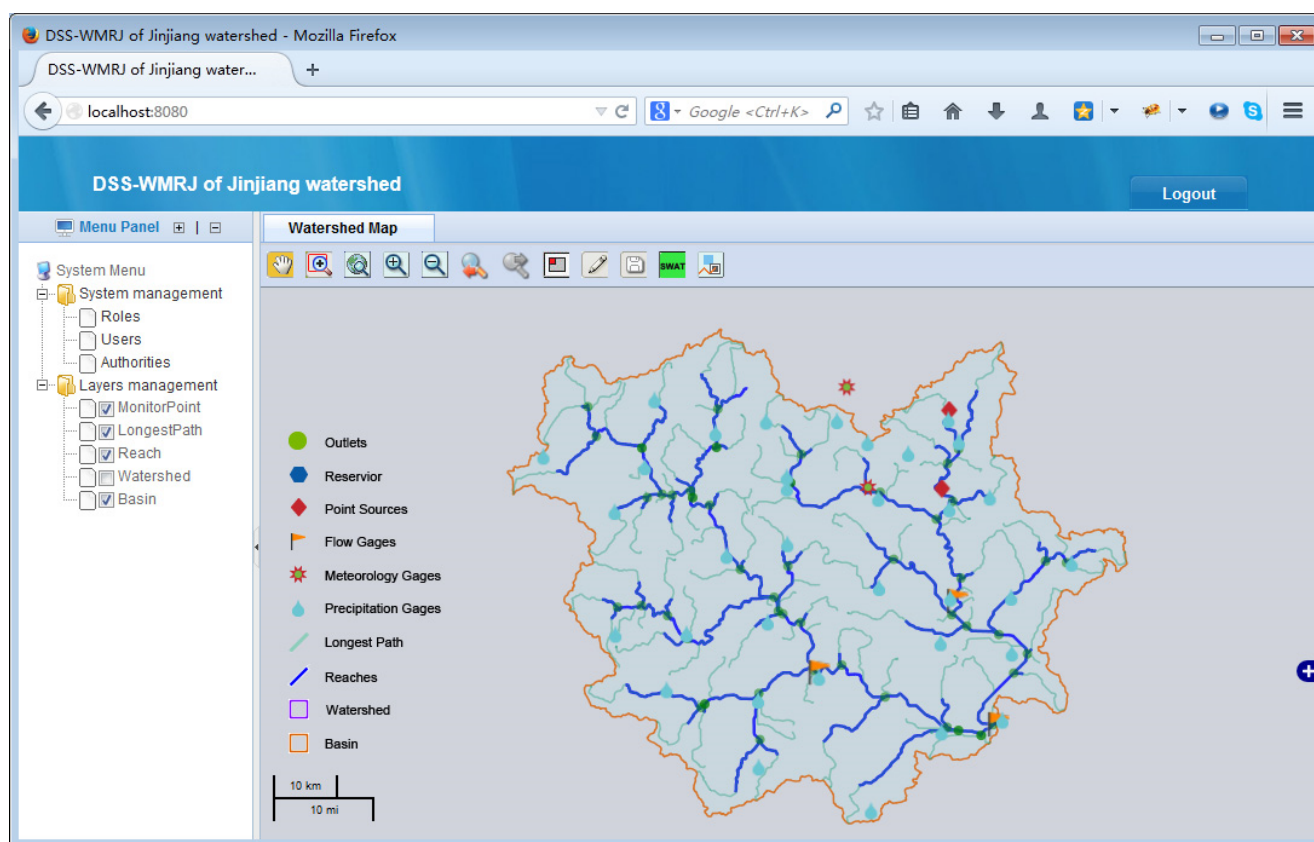

Figure 3. Main interface of DSS-WMRJ.

For a prototype of DSS-WMRJ, we only developed a tool to evaluate the soil and water conservation effect of a vegetation filter strip (VFS-Tool), which is a widely-used conservation practice to remove agricultural and urban pollutants before they reach nearby water bodies by establishing a strip of dense vegetative filter around the upslope pollutant sources. The interfaces of the VFS-Tool (Figure 4) are intuitive and easy to use. Users just need to click the tool icon in the toolbar, enter or select certain specific inputs that pertain to VFS and submit these inputs to the server.
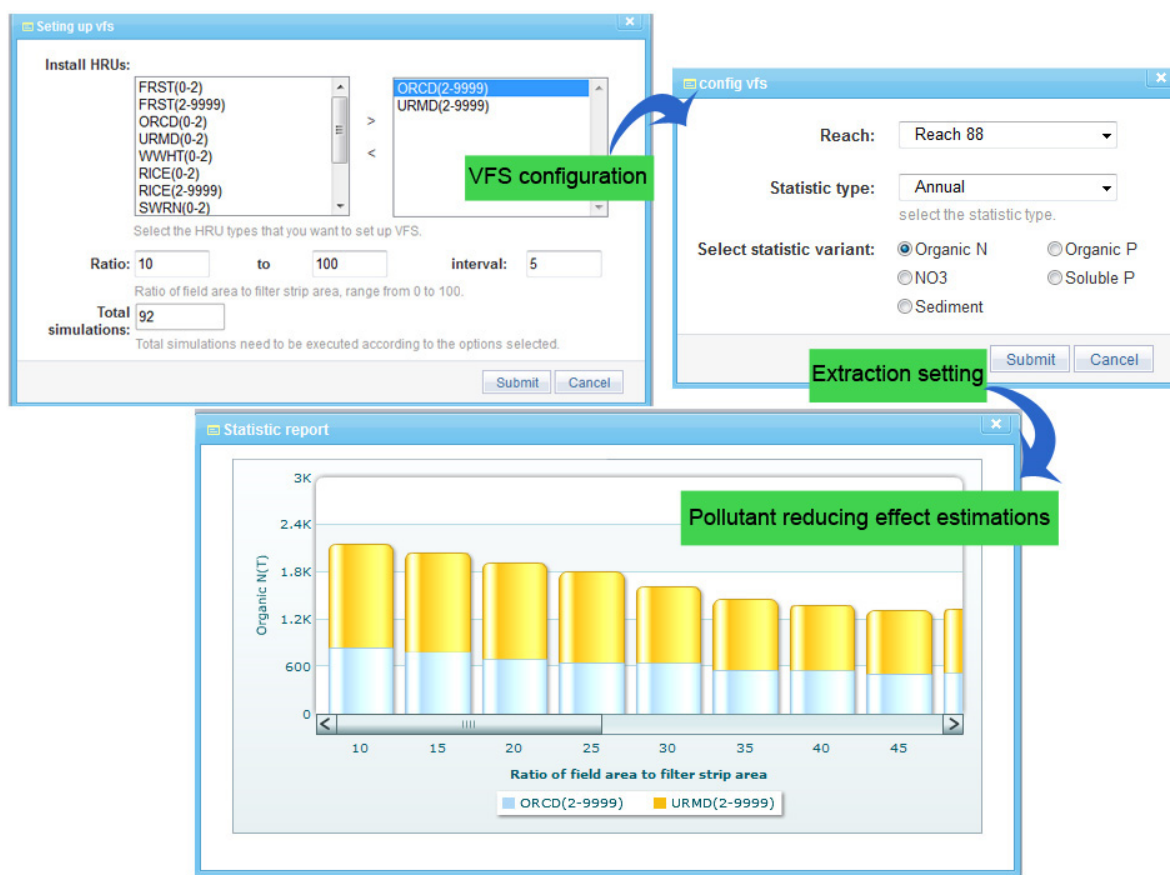
**Figure 4.** Interfaces of the vegetation filter strip (VFS)-Tool.

3.3. Performance Tests

The model simulation service is a key component of DSS-WMRJ that directly determines the achievement of real-time or quasi real-time decision making. The performance of this component is tested and evaluated. To perform the tests, one management scenario was used, which established VFSs around two kinds of HRUs (whose land use type is orchard or urban with a slope greater than two degrees) with varying ratios of field area to filter strip area (from 10 to 100 at an interval of five; Figure 4). The management scenario generated a total of 92 model simulations, which needed 110.4 min to finish if running the model in series, as each simulation took about 1.2 min. We will not go into the details about the pollution-reduction effect of VFSs, as our main objective here was to demonstrate the performance of the model simulation service. To evaluate the scalability of the model simulation service, the management scenario was performed in a Hadoop cluster (private cloud) with different number "of TaskTrackers (from one to eight), and each TaskTracker was allowed to perform four tasks simultaneously. These TaskTrackers are virtual machines on two physical servers. The configurations of the virtual machines are identical, and so are the physical ones, with the configuration details being listed in Table 1.

Table 1. The configurations of virtual and physical machines.

Virtual Machines	Physical Machines
4 logical processors Model name: Intel(R) Xeon(R) CPU E5520 @ 2.27 GHz CPU MHz: 2,261.060 RAM: 4 GB OS: 64-bit Red Hat Enterprise Linux 5.4	2 physical processors 4 cores for each physical processor 16 logical processors Model name: Intel(R) Xeon(R) CPU E5520 @ 2.27 GHz CPU MHz: 2,261.060 RAM: 16 GB VM management software: VMware ESXi 4.1

Figure 5 shows the results of the tests. The simulation time decreased as the number of TaskTrackers increased from one to eight. When eight TaskTrackers were used, the simulation time reached the lowest point (about 4.4 min). The number of TaskTrackers and the number of tasks in each TaskTracker are the major factors that affect the performance of the model simulation service (other factors are ignorable). As the task number in each TaskTracker was set to a constant value of four, the number of TaskTrackers becomes the only determinant factor that is negatively and nonlinearly proportional to the simulation time. Thus, we chose the inverse first order equation to generate a fit curve of the simulation time *vs.* the number of TaskTrackers (Figure 5). According to the trend of the fit curve, we believe that the lowest simulation time that could be achieved is about 2.14 min by using 23 TaskTrackers, as the simulation job cannot further parallelize beyond this number.

$$f(x) = 26.50/x + 0.9922$$
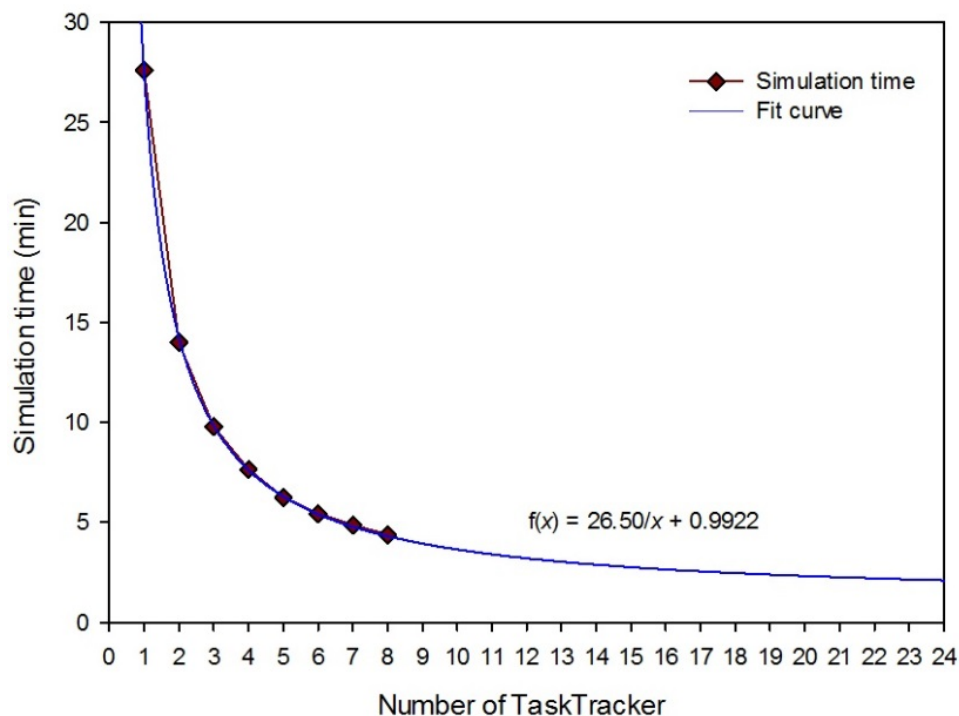
Figure 5. The performance of the model simulation service.

We also evaluated our model simulation service of DSS-WMRJ by comparing it with a widely-used SWAT auto-calibration tool (SWAT-CUP), which operates on a PC (Table 1). SWAT-CUP took 97.9 min to finish 92 simulations (not including the post-processing to gather information in order to generate a management report), while our service only took 4.4 min when running on eight TaskTrackers (each allowed four tasks running simultaneously). Although it is not a stringent comparison, as these two tools run on different environments, it still provided some convincing results that our model simulation service substantially reduced the execution time by parallelizing the model simulations on Hadoop clusters and, therefore, is able to support decision making with a reasonable amount of simulation time.

4. Discussion

The most outstanding features of DSS-WMRJ are its quasi-real-time decision making support, intuitive and wizard-style interfaces and excellent scalability. The implementation and test results showed that DSS-WMRJ can meet the goals of achieving intuitive and concise interfaces and supporting real-time or quasi real-time decision making. Besides, it is scalable, as the users just need to add more computing machines to the Hadoop cluster to scale up the system and achieve the goal of reducing the model simulation time. Other components of DSS-WMRJ, such as the map service and watershed management components, can also be scaled up by deploying machines and a simple configuration of Nginx.

Building on open source software and libraries is another valuable feature of DSS-WMRJ worthy of note (except the commercial chart component FusionCharts, due to its dynamic and excellent chart functionalities, but this component can be replaced by an open-source one). This feature makes it economic, as software license costs and other costs are not a factor, making it applicable to other watersheds. However, open-source software has some disadvantages, such as a lack of rapid building-up of tools and technology support, which is usually available for commercial software. Under joint efforts of the open-source community, the gap between open source and commercial software is increasingly narrowing.

Our DSS-WMRJ also has some other advantages. For example, it is accessible at any time and from anywhere by using a browser via the Internet or Intranet; and it is beneficial for information sharing and cooperation between individuals or institutions; thus, these will prompt users to participate in the decision making processes. It is easy to maintain and upgrade, as the system is deployed on the server. Besides, the web-based nature makes it easy to scale up and adopt for a cloud environment. However, there are some disadvantages, too. Compared with the desktop-based DSS-WM, it is more difficult to develop the web-based application, as it involves more languages and technologies, and other details need to be carefully considered, such as communications between browsers and servers.

As indicated by many studies and practices, stakeholders have significant impacts on the success of developing IT projects or facilities. This is especially so in situations when the funders and users are different individuals or not even in the same organization. As stakeholders may have different interests, it is very important to identify and involve these stakeholders at an early stage of the system implementation. In our case, we have three major groups of stakeholders: The funders, watershed managers and public users, all focusing on different aspects of the DSS-WMRJ. The funders are concerned more about the effectiveness of DSS-WMRJ; the watershed mangers are focussed on the

conciseness of the interface and the efficiencies; while the public users worry about the ease of information sharing. To fulfill these requirements, technologies, such as Web GIS, distributed models and cloud technologies, and the agile software development methodology were adopted in the development of DSS-WMRJ, to promote adaptive planning, evolutionary development, early delivery and to encourage rapid and flexible responses to changes. We are currently at an initial stage of the development cycle, and the prototype of DSS-WMRJ that we provided is mainly for the purpose of demonstrating to and communicating with stakeholders, stimulating them to provide more specific and accurate system demands. Therefore, the proposed prototype is not a fully functional one; nevertheless, the DSS-WMRJ will evolve into a fully-fledged tool.

In the future versions, DSS-WMRJ will be improved by a continuous enriching of the watershed management functionalities. The performance of DSS-WMRJ will be focused on, as well. For example, large hydrological models may take hours for a single execution, and this inevitably impedes the goal of real-time decision making support. This problem cannot be solved by simply paralleling the model simulations. It is important to reduce the model's execution time, so as to achieve the goal of real-time decision making. The hydrological processes at HRUs (the most process-intensive parts) and the sub-basin level are independent of each other by design in the modeling concept of SWAT. These processes at HRUs and sub-basins are traditionally computed in a serial manner by a single computer, which requires much computing time. Thus, parallelizing the calculation procedures for HRUs and sub-basins should be effective at reducing the simulation time, as proven by the studies of Yalew *et al*. [21] and Wu *et al*. [22], by using grid computing. Another possible solution to reduce the simulation time of SWAT is to divide the single and large SWAT watershed models into smaller ones and route them from the headwater basins to the terminal basin, then parallelize the calculation procedures of the headwater basins. Recently, Sun *et al*. [23] developed three metamodels (model reduction) to support real-time decision making regarding activities relative to surface water quality in a coastal watershed in Texas, USA. They approximated the SWAT model by a reduced order model in order to speed up the running time in the web environment. We would like to evaluate these two methods in our cloud environment and analyze the trade-offs for them.

Effect of management practices (such as EVFS) are not evaluated in our initial prototype of DSS-WMRJ. Many other studies [24–27] have already proven these management practices to be effective. We will also evaluate the incorporation of management practices in our future version of DSS-WMRJ with a well-calibrated SWAT watershed model. In addition, Hadoop technology is available via Amazon's Elastic MapReduce and Microsoft's HDInsight, thus making it possible to migrate DSS-WMRJ to a public cloud. We will evaluate the model simulation service with one of these public services.

5. Conclusions

A user-friendly and quasi-real-time prototype of DSS-WMRJ was developed by seamlessly integrating an open-source Web GIS tool, Geoserver, a modeling component, SWAT, a cloud computing platform, Hadoop, and other open-source components and libraries. Due to its flexible and innovative features, DSS-WMRJ has some advantages over other decision support systems for watershed management: (1) Quasi-real-time decision making is obtained by utilizing cloud computing technology;

(2) An intuitive and user-friendly GUI is provided, which largely enhances the user experience; and (3) It is very economic, as the DSS-WMRJ was almost entirely built on open-source software, and this feature lends to it great prospects of being applied to other watersheds. This is also valuable and informative for building other environmental DSSs.

However, as a prototype of DSS-WMRJ, there are some inadequacies (e.g., the nutrient components of SWAT were not well calibrated and evaluated, due to insufficient monitoring data, and limited management options and functionalities were implemented), and thus, continuous improvement is necessary. In the next version of DSS-WMRJ, more management practices will be incorporated, and the model simulation time will be further reduced by modifying the structure of the SWAT model. We will also evaluate DSS-WMRJ with a well calibrated (including the runoff, sediment and nutrients) model and evaluate the model simulation service with a public cloud, such as Amazon's cloud services.

Acknowledgments

Author Contributions

Xingwei Chen had the basic idea of the present work and coordinated the research group. Dejian Zhang contributed in designing, developing and testing the prototype web-based decision support system for watershed management and applying it to the case study. Dejian Zhang initiated the manuscript writing, Huaxia Yao made improvements to the English writing.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Karimi, P.; Bastiaanssen, W.G.M.; Molden, D. Water accounting plus (wa plus)—A water accounting procedure for complex river basins based on satellite measurements. *Hydrol. Earth Syst. Sci.* **2013**, *17*, 2459–2472.
2. Chung, E.-S.; Lee, K.S. Prioritization of water management for sustainability using hydrologic simulation model and multicriteria decision making techniques. *J. Environ. Manag.* **2009**, *90*, 1502–1511.
3. Qi, H.H.; Altinakar, M.S. Vegetation buffer strips design using an optimization approach for non-point source pollutant control of an agricultural watershed. *Water Resour. Manag.* **2011**, *25*, 565–578.
4. Wool, T.A.; Davie, S.R.; Rodriguez, H.N. Development of three-dimensional hydrodynamic and water quality models to support total maximum daily load decision process for the neuse river estuary, north carolina. *J. Water Res. Pl-Asce* **2003**, *129*, 295–306.

5. Park, Y.S.; Park, J.H.; Jang, W.S.; Ryu, J.C.; Kang, H.; Choi, J.; Lim, K.J. Hydrologic response unit routing in swat to simulate effects of vegetated filter strip for south-korean conditions based on vfsmod. *Water* **2011**, *3*, 819–842.

6. Kim, Y.-J.; Kim, H.-D.; Jeon, J.-H. Characteristics of water budget components in paddy rice field under the asian monsoon climate: Application of hspf-paddy model. *Water* **2014**, *6*, 2041–2055.

7. Matthies, M.; Giupponi, C.; Ostendorf, B. Preface—Environmental decision support systems: Current issues, methods and tools. *Environ. Model. Softw.* **2007**, *22*, 123–127.

8. McIntosh, B.S.; Ascough, J.C.; Twery, M.; Chew, J.; Elmahdi, A.; Haase, D.; Harou, J.J.; Hepting, D.; Cuddy, S.; Jakeman, A.J.; *et al*. Environmental decision support systems (EDSS) development—Challenges and best practices. *Environ. Model. Softw.* **2011**, *26*, 1389–1402.

9. Cortes, U.; Sanchez-Marre, M.; Ceccaroni, L.; Roda, I.R.; Poch, M. Artificial intelligence and environmental decision support systems. *Appl. Intell.* **2000**, *13*, 77–91.

10. Mysiak, J.; Giupponi, C.; Rosato, P. Towards the development of a decision support system for water resource management. *Environ. Model. Softw.* **2005**, *20*, 203–214.

11. Cau, P.; Paniconi, C. Assessment of alternative land management practices using hydrological simulation and a decision support tool: Arborea agricultural region, sardinia. *Hydrol. Earth Syst. Sci.* **2007**, *11*, 1811–1823.

12. Hipel, K.W.; Fang, L.; Kilgour, D.M. Decision support systems in water resources and environmental management. *J. Hydrol. Eng.* **2008**, *13*, 761–770.

13. Rao, M.; Fan, G.; Thomas, J.; Cherian, G.; Chudiwale, V.; Awawdeh, M. A web-based GIS decision support system for managing and planning USDA's conservation reserve program (CRP). *Environ. Model. Softw.* **2007**, *22*, 1270–1280.

14. Zeng, Y.; Cai, Y.; Jia, P.; Jee, H. Development of a web-based decision support system for supporting integrated water resources management in Daegu city, South korea. *Expert Syst. Appl.* **2012**, *39*, 10091–10102.

15. Sun, A. Enabling collaborative decision-making in watershed management using cloud-computing services. *Environ. Model. Softw.* **2013**, *41*, 93–97.

16. Lam, C. *Hadoop in Action*, 1st ed.; Manning Publications Co.: Stamford, CT, USA, 2010; p. 336.

17. White, T. *Hadoop: The Definitive Guide*, 2nd ed.; O'Reilly Media: Sebastopol, CA, USA, 2009; p. 625.

18. Arnold, J.G.; Fohrer, N. SWAT2000: Current capabilities and research opportunities in applied watershed modelling. *Hydrol. Process.* **2005**, *19*, 563–572.

19. Arnold, J.G.; Srinivasan, R.; Muttiah, R.S.; Williams, J.R. Large area hydrologic modeling and assessment part I: Model development1. *JAWRA* **1998**, *34*, 73–89.

20. Wang, L.; Chen, X.W. Runoff simulation with calibration and validation of three station in Jinjiang river basin. *Sci. Soil Water Conserv.* **2007**, *5*, 21–26. (In Chinese).

21. Yalew, S.; van Griensven, A.; Ray, N.; Kokoszkiewicz, L.; Betrie, G.D. Distributed computation of large scale SWAT models on the grid. *Environ. Model. Softw.* **2013**, *41*, 223–230.

22. Wu, Y.P.; Li, T.J.; Sun, L.Q.; Chen, J. Parallelization of a hydrological model using the message passing interface. *Environ. Model. Softw.* **2013**, *43*, 124–132.

23. Sun, A.Y.; Miranda, R.M.; Xu, X. Development of multi-metamodels to support surface water quality management and decision making. *Environ. Earth Sci.* **2015**, *73*, 423–434.

24. Chu, T.-W.; Lin, Y.-C.; Shirmohammadi, A.; Huang, Y.-C. BMP evaluation for nutrient control in a subtropical reservoir watershed using SWAT model. In *Proceedings of the 2013 the International Conference on Remote Sensing, Environment and Transportation Engineering*, (Rsete 2013); Gahegan, M.N., Xiong, N., Eds.; Atlantis Press: Paris, France, 2013; Volume 31, pp. 914–917.

25. Liu, R.; Zhang, P.; Wang, X.; Wang, J.; Yu, W.; Shen, Z. Cost-effectiveness and cost-benefit analysis of BMPs in controlling agricultural nonpoint source pollution in China based on the SWAT model. *Environ. Monit. Assess.* **2014**, *186*, 9011–9022.

26. Giri, S.; Nejadhashemi, A.P.; Woznicki, S.; Zhang, Z. Analysis of best management practice effectiveness and spatiotemporal variability based on different targeting strategies. *Hydrol. Process.* **2014**, *28*, 431–445.

27. Dechmi, F.; Skhiri, A. Evaluation of best management practices under intensive irrigation using swat model. *Agric. Water Manag.* **2013**, *123*, 55–64.