

Software Reuse Methods to Improve Technological Infrastructure for e-Science

James J. Marshall¹, Robert R. Downs², Chris A. Mattmann^{3,4}

¹*INNOVIM
NASA Goddard Space Flight Center
Mail Stop 614.9
Greenbelt, Maryland 20771 USA
James.J.Marshall@nasa.gov*

²*Center for International
Earth Science Information Network
Columbia University
Palisades, NY 10964 USA
rdowns@ciesin.columbia.edu*

³*Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA
mattmann@jpl.nasa.gov*

⁴*Computer Science Department
University of Southern California
Los Angeles, CA 90089, USA
mattmann@usc.edu*

Abstract

Social computing has the potential to contribute to scientific research. Ongoing developments in information and communications technology improve capabilities for enabling scientific research, including research fostered by social computing capabilities. The recent emergence of e-Science practices has demonstrated the benefits from improvements in the technological infrastructure, or cyberinfrastructure, that has been developed to support science. Cloud computing is one example of this e-Science trend. Our own work in the area of software reuse offers methods that can be used to improve new technological development, including cloud computing capabilities, to support scientific research practices. In this paper, we focus on software reuse and its potential to contribute to the development and evaluation of information systems and related services designed to support new capabilities for conducting scientific research.

Keywords: Cloud computing; open source software; scientific computing; software reuse

1. Introduction

Scientific practices have improved as communities have adopted new information and communications technology (ICT). Recognizing changes in scientific practices that have evolved from the use of new ICT during the past decade, new terms such as “cyberinfrastructure” and “e-Science” have emerged to describe the technological infrastructure designed to

improve capabilities for conducting scientific research. Social computing, as well as other practices that have the potential to contribute to scientific research, have been enabled by these developments.

As technology has evolved, the methods and tools of software reuse have become more relevant for the deployment of new systems to support scientific research. Employing software reuse techniques can enable the adoption of new technological approaches to support new models for conducting scientific research. Software reuse methods and tools also can contribute to the evaluation of systems being developed to support scientific research. Software reuse instruments, such as the Reuse Readiness Levels (RRLs) [1], offer the potential to improve capabilities for software development and evaluation when developing and adopting software assets for scientific support systems [2].

In addition to improving methods for developing and evaluating new systems to support scientific research, employing software reuse when adopting new technological approaches also has the potential to improve practices for software reuse. Similar to the participatory practices of open source software development communities, adopters of various artifacts, including methods and instruments, could contribute to their improvement, based on experiences using such artifacts [3].

Recent developments in technological infrastructure that supports scientific research include cloud computing, which offers capabilities for research communities to utilize scientific data and services to conduct analyses that otherwise would be more costly or prohibitive to complete. Cloud computing is a prime example of a technology that can benefit from software reuse.

Cloud computing and its potential to support scientific research is introduced in section 2. Improvements in the reusability of software that are applicable to cloud computing are reviewed in section 3. Reuse Readiness Levels are described in section 4, in terms of their applicability to the development of cloud computing software. The use of the Reuse Readiness Levels to evaluate cloud computing capabilities for scientific research is described in section 5. The potential to contribute to the improvement of the Reuse Readiness Levels, based on experiences in cloud computing, is described in section 6 and a summary is presented in section 7. Note that while this position paper focuses on cloud computing as an example, the software reuse tools and concepts presented here also can be applied to other areas of technological development to support scientific research.

2. Cloud Computing Support for Scientific Research

Cloud computing offers the potential to improve capabilities for enabling scientific research. In cloud computing environments, software services focused on computation and storage are provided to users, who can represent various organizations and interests. Using cloud computing to provide data and services to scientists and their teams offers a variety of benefits for the scientific research community, at the very least the ability to “pay to play” and effectively purchase scalability at a commodity cost. Similar to the benefits that cloud computing offers for other industries that provide or use cloud services [4], cloud computing also can improve software and systems development capabilities for scientific research.

Additionally, if the use of cloud services are managed properly, the adoption of cloud computing can reduce the costs of providing software services [5]. Attaining cost-efficient capabilities for providing cloud computing services also could benefit scientific research communities, depending on the match between scientific computing needs, cost structures, and performance capabilities of selected cloud platforms [6].

Cloud computing offers the potential to improve capabilities for developing, maintaining, and disseminating scientific research data. Economies of scale can be attained by using cloud computing to provide services for sharing biomedical data [7]. Bell, Hey, and Szalay [8] suggest that the rate of progress to be achieved in data-intensive e-Science will depend on the ability of scientific disciplines to adopt cloud computing and other emerging technologies.

Furthermore, employing open source software development practices can improve software reuse capabilities [9], and also offers possible advantages for

cloud computing [4]. While there are risks in adopting open source software development practices, strategies can be deployed to mitigate such risks and include dedicating needed resources early, initiating pilot projects to learn about open source development, increasing awareness and visibility of open source software, and obtaining top management support for open software development [10].

3. Software Reuse Methods Applicable to Cloud Computing for Scientific Research

Research on the reusability of software has improved software development practices by taking a long-term perspective to software development. Rather than considering a software application or system as a product to be deployed for just an immediate purpose, software reuse research has adopted a perspective that enables the continuing use of software and related assets beyond their initial purpose. The benefits of software reuse can be realized by the original developers or by the new adopters.

Enabling the reuse of applications that run locally for reuse in the cloud can facilitate the adoption of cloud computing for scientific research and the transfer of local operational services to the cloud. Moving scientific data services between a private cloud and a public cloud may address cost concerns of cloud computing for scientific research [11]. Preparing scientific support services for reuse would enable organizations to address the potential need for an increase in demand for services without excessive lead time.

Taking a systematic approach to the development of systems, applications, and services for potential reuse could enable providers and developers of scientific systems to port their services to one or more cloud providers to increase capacity as the needs to serve user communities evolve. Since different cloud providers may use different standards or conventions, the interoperability of the service needs to be considered when porting it to new or additional cloud providers. Modifications to the original service may be necessary to ensure that it runs properly on each new provider’s system.

Establishing a reference architecture also could facilitate the development of scientific research services for use in the cloud. Creating a reference architecture for cloud services to reuse processes and solutions fosters collaborative provision of services as well as sharing among users [12].

4. Reuse Readiness Levels to Guide Cloud Computing Software Development

Recent developments in software reuse have included NASA’s Reuse Readiness Levels (RRLs) [1], which can

be used to improve the creation, evaluation, and maintenance of software systems, components, and other assets. RRLs offer the potential to contribute to the development of cloud computing systems and software services established to support scientific research communities [13]. Developed by the Earth Science Data Systems (ESDS) software development community and described in [14], the RRLs were created to assist scientific data systems professionals in the design and development of reusable software by facilitating a systematic approach to the development process [15].

The RRLs are comprised of nine different topic areas. Alphabetically, they are “documentation, extensibility, intellectual property issues, modularity, packaging, portability, standards compliance, support, and verification and testing” [16]. Each of the topic areas was selected based on its potential use for assessing the reuse maturity of software assets. In addition, each of these topic areas consists of up to nine levels, providing a grid that can be used to measure specific aspects of software maturity with the aim of assessing and improving its reusability. These measures were designed without any specific type of software or system in mind. As such, they are as applicable to the cloud computing environment as to traditional software and systems development environments.

Alternatively, the nine overall RRLs summarize the topic area levels in a more compact form, offering capabilities to assign a simple value to evaluate software reuse maturity. Assigned ratings can range from limited reusability, when the software is not being recommended for reuse at RRL 1, to demonstrated extensive reusability, when the software is being reused by many classes of users over a wide range of systems at RRL 9. In contrast to using the entire grid for assessing software, the overall RRLs enable a single rating, which also fosters a quick-look at reusable assets. The overall RRL score can be computed via a weighted average of score assignments in each of the topic areas. The potential reuser determines the weights and scores in terms of the intended purpose [14].

In general, employing the entire grid of topic area level assessment criteria will be more useful to software developers looking to reuse existing assets and/or to make their own assets more reusable.

5. Evaluating Cloud Computing Capabilities for Scientific Research

In addition to contributing to the development of software for potential reuse, the RRLs also can be used to evaluate the potential reuse of software and system components developed to support scientific data systems [2]. As noted in the previous section, the RRL topic area levels provide a wealth of information regarding the

maturity of software assets in nine different areas related to reusability. Software developers seeking to reuse existing assets can use these topic area levels to make assessments of candidate software that could be suitable for their needs. By assessing each candidate software asset in each of the nine topic area levels, developers can get a good sense of how ready each asset is to be reused. Assets that receive higher scores will be more likely to be reusable than those with lower scores.

Likewise, these assessments provide information about what types of modifications would be necessary in order to adapt an asset for use in the intended, new context. Looking at the topic area levels, developers can determine the minimum scores required for an asset to be reused in the context of the new system under development. Since it is likely that not all assets will meet all of these minimum requirements, the developers need to evaluate the shortcomings of the candidate assets. For each topic area level, the difference between the assessed level and the required level provides information about what types of modifications would be necessary to achieve the minimum level needed for reuse in the new system. Developers can review these differences and determine the amount of development necessary for each candidate asset to be considered for adoption. Identifying such gaps enables developers to determine which assets, if any, would be best suited for modification and reuse in the new system. In addition, the priority that the developers assign to a particular topic area also could be used when determining the suitability of particular software assets for reuse.

For example, consider a case where portability of the existing reusable software is a key factor in whether it can be reused in the development of a new system. Perhaps the new system is running on one operating system or platform, but the ideal candidate reusable software was written for another operating system or platform. The developers may determine that a portability topic area level of 6, where the software is portable, would be necessary to adopt the candidate software for their system. In contrast, a candidate asset that scores 3, where the software is only portable with significant costs, would not be suitable for reuse in that state. However, the developers can evaluate the effort required to raise the asset’s score from 3 to 6, and this knowledge would factor in to their decision as to the readiness of this asset to be reused in their context. Assets that require the least amount of modifications to be reused in the new context are the ones that are most likely to be reused, and the RRLs can help developers determine how many modifications are necessary, and of what types, for each candidate reusable software asset that is being considered.

6. Improving Software Reuse Methods from Cloud Computing Experiences

In addition to the potential for RRLs to contribute to the cloud computing infrastructure that supports e-Science, experiences using the RRLs for the development of cloud computing services to support scientific research also could contribute to the improvement of the RRLs. Feedback from the community regarding their experiences with practical applications of the RRLs can provide vital information on the current state of these efforts. Assessments of actual reusable software assets can uncover areas for improvement, such as discovering where two levels are defined too closely to truly distinguish each, or when levels are not sufficiently quantitative to allow a relatively unambiguous assessment. Specific suggestions for simplifying the definitions for topic area levels also could reduce the time required to evaluate candidate assets.

Experiences using the RRLs in the development and use of cloud computing capabilities also could demonstrate where additional precision or further explanation is needed for levels within the RRLs. While some differences in the results of assessments of a given asset by different people are expected, these should be small for an assessment instrument that is reliable. If differences among reviewer assessments are large, such differences could be an indicator of where the precision of the measures would need to be refined to improve the reliability of the RRLs as an instrument for evaluating the potential reusability of systems, software, and other computing assets. As cloud computing continues to grow, input from this community will be important in ensuring that the RRLs are relevant and applicable to the software and systems developers working in this area and the software assets that they create.

7. Summary

In this paper, we have described how software reuse methods have the potential to improve new developments in information and communications technology (ICT) that are being established to enhance capabilities for scientific communities to conduct research. Like other advancements in research practices, social computing also benefits from ICT. Cloud computing has been shown as an example of ICT that can enhance e-Science capabilities. Lessons learned from software reuse research and practices can contribute to the development of cloud computing. Likewise, developments in cloud computing can contribute to the improvement of software reuse tools and techniques.

The applicability of the Reuse Readiness Levels (RRLs), for improving the development and evaluation of cloud computing services, also has been described. While use of the RRLs has the potential to improve capabilities for evaluating software assets for reuse in cloud computing, feedback from such use also has the potential

to improve these measures for reuse in other contexts. Just as cloud computing can benefit from and contribute to software reuse methods and tools, such as the RRLs, other ICT development efforts also can benefit from and contribute to software reuse. Building on these and similar efforts can ultimately contribute to developments in cyberinfrastructure to support e-Science.

8. Acknowledgments

The authors are grateful to the members of the National Aeronautics and Space Administration (NASA) Earth Science Data Systems Software Reuse Working Group who have contributed to the efforts described in this work. Support was provided for Robert Downs under NASA contract NNG08HZ11C. This effort was supported in part by the Jet Propulsion Laboratory, managed by the California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

9. References

- [1] NASA Earth Science Data Systems Software Reuse Working Group, "Reuse Readiness Levels (RRLs), Version 1.0," 30 April 2010. Available online, URL=<http://www.esdswg.org/softwarereuse/Resources/rrls/>
- [2] R.R. Downs and J.J. Marshall, "A Proposal on Using Reuse Readiness Levels to Measure Software Reusability", *Data Science Journal*, CODATA, July 2010, pp. 73-92.
- [3] G. Fischer, "End-user development and meta-design: Foundations for cultures of participation", *Journal of Organizational and End User Computing*, January-March 2010, pp. 52-82.
- [4] M. Armbrust, A. Fox, R. Griffith, A. Da. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. "A View of Cloud Computing: Clearing the clouds away from the true potential and obstacles posed by this computing capability", *Communications of the ACM*, ACM, New York, NY, April 2010, pp. 50-58.
- [5] M.D. de Assunção, A. di Costanzo, and R. Buyya, "Evaluating the cost-benefit of using cloud computing to extend the capacity of clusters", In *Proceedings of the 18th ACM international symposium on High performance distributed computing (HPDC '09)*, ACM, New York, NY, 2009, pp. 141-150.
- [6] G. Juve, E. Deelman, K. Vahi, G. Mehta, B. Berriman, B.P. Berman, and P. Maechling, "Data Sharing Options for Scientific Workflows on Amazon EC2", In

Proceedings of the 2010 ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC '10), IEEE Computer Society, Washington, DC, 2010, pp. 1-9.

[7] A. Rosenthal, P. Mork, M.H. Li, J. Stanford, D. Koester, and P. Reynolds, "Cloud computing: A new business paradigm for biomedical information sharing", *Journal of Biomedical Informatics*, Elsevier, San Diego, CA, April 2010, pp. 342-353.

[8] G. Bell, T. Hey, and A. Szalay, "Beyond the Data Deluge", *Science*, AAAS, New York, NY, 6 March 2009, pp. 1297-1298.

[9] P. Vitharana, J. King, and H.S. Chapman, "Impact of Internal Open Source Development on Reuse: Participatory Reuse in Action", *Journal of Management Information Systems*, M.E. Sharpe, Fall 2010, pp. 277-304.

[10] Ø. Hauge, D.S. Cruzes, R. Conradi, K.S. Velle and T.A. Skarpenes, "Risks and Risk Mitigation in Open Source Software Adoption: Bridging the Gap between Literature and Practice", In *Open Source Software: New Horizons*, Springer, Boston, MA, 2010, pp. 105-118.

[11] J. Becla, K. Lim, and D.L. Wang. "Report from the 3rd Workshop on Extremely Large Databases", *Data Science Journal*, CODATA, February 2010, pp. MR1-MR16.

[12] H.R. Motahari-Nezhad, J. Li, B. Stephenson, S. Graupner, and S. Singhal, "Solution Marketplace for Service Composition and Integration", In *3rd International Workshop on Web Service Composition and Adaptation*, Los Angeles, CA, 2009, Available online, URL= <http://www.hpl.hp.com/techreports/2009/HPL-2009-96.pdf>

[13] J.J. Marshall, R.R. Downs, and C.A. Mattmann, "Progress Towards a NASA Earth Science Reuse Enablement System (RES)", In *Proceedings of the 2010 IEEE International Conference on Information Reuse and Integration*, IEEE, Las Vegas, NV, 2010, pp. 340-343.

[14] J.J. Marshall and R.R. Downs, "Reuse Readiness Levels as a Measure of Software Reusability", In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium 2008*, IEEE, Boston, MA, 2008, pp.III-1414-III-1417.

[15] J.J. Marshall, R.R. Downs, and S. Samadi, "Building the Next Generation of Aerospace Data Processing Systems by Reusing Existing Software Components", In *Aerospace Technologies Advancements*, IN-TECH, Vukovar, Croatia, 2010, pp. 25-36.

[16] J.J. Marshall, R.R. Downs, S. Samadi, N.S. Gerard, and R.E. Wolfe, "Software Reuse to Support Earth Science", *Journal of Frontiers of Computer Science and Technology*, Ministry of Industry and Information of PRC, Beijing, China, 2008, pp. 296-310.