

Reuse of software

Technology, Information Technology



Reuse of software is the utilization of existing assets in some kind within the process of software product development. This does not only apply to codes, but to components of the lifecycle of software development. These include designs, software components, documentation, and test suites. Therefore, software reuse has been reported to increase productivity, to save time, and to reduce software development cost. In the past, reuse of software was about reusing entire applications. However, currently, reuse of software involves the redeployment of components across the complete lifecycle of development beginning with domain modeling and specifications, design of software, coding, testing, to operation and maintenance, (Nazareth and Rothenberger, 2004). However, cost savings from reusing existing software is not simply proportional to the size of the components that are used. Other factors also affect the cost. When components of existing software are reused, they might bring several problems. For example, compatibility of the reused components may not conform to the new developed system. Also, according to Vasantha and Jasmine (2008), they might bring additional bugs in the communication between the components after integration. As a result, component maintenance costs arise. When software is reused, it is the costs of application maintenance that are reduced. However, maintenance costs of components can become quite huge. The reason is that the component has to respond to the various requirements of the various applications that run in varying environments. These applications have differing reliability requirements, and some might require a different level of maintenance support. Therefore, additional costs arise in the form of maintenance support. When a component is reused, the cost savings gained are those

that could have been used in developing a new component. This means that the more robust the component, the greater the cost savings. However, other costs of reuse arise in finding and understanding components that can be used for replacement. For example, are the other parts of the system capable of accommodating the reused component? Are they compatible? When it comes to small components, the reuse costs might surpass the development-from-scratch costs. This means that the costs savings are substantial when dealing with a more complex system. This means that reuse costs are not only proportional to the size of the component, but also depend on other factors such as system complexity and function.

References

Component-Based Software Engineering [online]. (2012). Available from: . [Accessed 14 Dec 2012].

A Generalized Reuse Framework for System Development [online]. (2012). Available from: . [Accessed 13 Dec 2012].

Nazareth, L. D. and Rothenberger, M. A. 2004. Assessing the Cost-Effectiveness of Software Reuse. *Journal of Systems and Software*, Vol. 73, No. 2, pp. 245-255.

Sommerville, I. 2010. *Software Engineering*. New Jersey: Pearson Books.

Vasanth, R. and Jasmine, K. S. 2008. Cost Estimation Model for Reuse Based Software Products. *Proceedings of the International Multi-Conference of Engineers and Computer Scientists*, 2008, Vol. 1, pp. 1-4.