

Countermeasures for the Implementation of Information Management System Development of Computer Software

Miao Xianghui

Dalian Vocational and Technical College, Dalian City, Liaoning Province, 116035, China

Keywords: Development management system, Web application program, Design framework

Abstract: Nowadays, software development information management systems is facing more challenges than ever due to the inherent high complexity and the growing need for a fast service-ready paradigm. At the same time, as the developers need higher-quality professional software systems, however, no sufficient guidance provided by software engineering methods and standards, the rapid development of qualified commercial software cannot get support. For the above problem, this paper discusses the advantages of software development, the pattern-based software framework and the corresponding system design framework for the rapid development of Web applications to verify the benefits, and then provide some reference value for the implementation of computer software development management system. In this paper, it introduces a framework based on the information pattern of computer software development, and the corresponding design framework for the rapid development of Web applications. The goal of the framework is that, through the software patterns, developers should be able to separate development efforts from business rules, thereby reducing problems caused by developers' lack of business experience. Through the modularity and standardization, the evolutionary robustness of the system can be easily achieved.

1. Introduction

1.1 Object-Oriented Technology

Traditional computer software development often leads to the emergence of custom systems that have significant advantage, which can adapt to the user's business model optimally, and thus can meet most of the requirements of internal proprietary knowledge or practices. But it also has the serious disadvantage of high cost and the limitation and dependence of individual expertise, which will lead to significant difficulties in maintenance and development, thus that reveals the necessity of object-oriented technology^[2].

Object-oriented (OO) technology is one modern software approach that 'echoes' the emergence, behavior and interaction of things in the real world to organize data and code^[3]. OO technology has greatly influenced the development and maintenance of computer software through faster development, cost savings and quality improvements.

For the OO technology, software module, one part of the software design and service, can actually be deployed, and treated as the separate part of the system. It rarely sells, buys, swaps, or deploys software objects in different projects. Another way in which module differ from object is that software module often involve more design issues, such as different implementation techniques, more complex function, and even languages and designs with specific process controls, that is, design patterns. Although 'module' and 'object' are often used interchangeably, when we talk about reuse, we must deal with them separately.

1.2 Xml

XML is a standard language supported by the World Wide Web Consortium (W3C). XML has many useful features, such as application neutrality (vendor independence), users extensibility, the ability representing arbitrarily complex information, validation schemes for data structures, and ease of reading.

The design goals of XML include the following: XML should be available directly on the Internet; XML will support a variety of applications; XML should be compatible with SGML; It should be easy to write programs that process XML documents; The number of optional features in XML should be kept to an absolute minimum, ideally it should be 0; XML documents should be clear and readable. XML design should be ready quickly. XML should be designed to be concise and clear; XML documents should be easy to create; Simplicity in XML tags is critical.

1.3 Design Mode

For the applications in variety of domains, there are certain characteristics of design activities that occur repeatedly. If we can make this knowledge explicit and publicly available to other practitioners, it is obviously important, the most important advantage of which is that the inexperienced designers have access to an instantly available technology library to make a good design live longer, which is especially important for the process of developing relevant design technical expertise. By inheriting object-oriented design technology, design patterns have become the main method by which expert designers solve the common and specific problems in design program.

Design pattern (DP) refers to a series of familiar usages and constructs used throughout the system design, allowing for rapid coding of certain components by following certain step patterns. By providing a clear specification of object interaction and its potential purpose, the documentation and maintenance of existing systems can be improved. One of the main purposes of design patterns is to help software engineers understand the common characteristics of software objects and module in their professional field.

2. Framework

Framework is another extraordinary software technology that involves both module reuse and design reuse. The software of any domain developed based on the existing systematic frame is highly reusable if the framework is considered in the process of design. Framework is a set of collaborative abstract and concrete classes, which are constructed as reusable components in certain domains, and contains the instance interface definitions. Designers can inherit from the instances of these classes to create new classes. Basically, the framework provides an environment to support all the activities involved in software development, however, you may run into problems when trying to create combinations of different framework.

Applying framework and design patterns to development systems can improve our productivity. Framework is usually used for domain-specific problems, while design patterns are used to solve general problems. If we find a design pattern that fits our requirements, we can apply the design pattern to any application domain, making the design pattern more generic than the framework.

3. Struts and Spring

Struts is one part of Apache Software Foundation program belonging to Jakarta, Struts is a flexible control layer based on standard technologies such as Java servlet, Java Bean, Resource Bundles, and XML. This framework can help programmers create extensible development environments for your applications based on published standards and proven design patterns. Struts is based on a concept called MVC, which derived from the Small Talk Model-view-controller framework. Under MVC, the application is divided into three different parts, the problem domain is represented as model and the output to the user is View, while the input from the user is represented as a Controller. Struts consists of a set of collaborative classes and components and JSP tag lib servlets. The model layer in an MVC application is usually project-specific and that framework is designed to simplify the process of accessing to the business side of the application, but leaves the programming part to other products, such as JDBC, Enterprise Java Beans etc.

Spring is a layered Java/J2EE application framework based on the code released by Johnson in expert one-to-one J2EE design and development⁴.

4. Feasible Solution for the Mode-Based Frame-Software Evolutionary Robustness

Computer networks and Internet-related business activities have become the primary medium for enterprise-level software deployment. Today, network operating environment greatly extends the scope of scalability from a few user activities to millions of simultaneous interactions. Extensibility means that ‘users can guarantee the service quality without changing the software’ because the number of users and connections increases indefinitely. Scalable system should be robust, reliable, flexible, and able to adapt to changing conditions, which are key characteristics of modern Web applications, besides, it should include the following functions: Openness-Be able to integrate with a variety of platforms and systems; Extensibility-Easy to extend its interface and application; Maintainability-Improves as the development of software and hardware technology; Modularity-Easy to plug in or integrate with other components; Distribution- Deployable to the Internet world.

In the past, companies have spent a lot of money on commercial software systems, many of which are still critical to the business. Due to many factors, including error correction, requirements change, business rule change, restructuring, etc., these legacy systems need to be maintained and developed. The basic problem of maintaining and developing legacy systems is to understand the subject system. It is expected to develop traditional systems into extensible systems, which can greatly reduce software costs and improve the performance of basic systems. However, a lack of consideration for evolutionary robustness from the beginning limits the effectiveness and efficiency of making changes to legacy systems. In the long run, through the tradeoff between rapid-declining new systems and old values, an appropriate framework is required. Apart from that, the software needs to be developed from the beginning based on the efficiency, using efficiency-pattern-based technology, to achieve robustness.

5. Conclusion

In order to implement the computer software development information management system, meanwhile, to respond to the needs of Web applications, many solutions have emerged, such as Struts, Spring framework etc., especially the upcoming evolution, to complete an well-organized system and reduce overall costs. In addition, the quality of software development systems can be guaranteed due to the promised rules and guidelines in a pattern-based framework.

Acknowledgment

This thesis is the outcome of scientific research innovation team construction of Dalian Vocational and Technical College in 2019.

References

- [1] Zhang Zhenzhong.Discussion on the Difficulties and Countermeasures of Computer Software Development and Design[J].Goods and Quality,2016,000(034):64-64,65.
- [2] Chenping.Study on the Difficulties and Countermeasures of Computer Software Development and Design[J].Computer Knowledge and Technology, 2017,013(006):94-95,98.
- [3] Zhang Yali.Problems and Countermeasures of Computer Software Development and Database Management [J].Information and Computer, 2016,000(010):194-195.
- [4] Li Du.Design and Implementation of Information Management System of Computer Software Development [J].Electronic Design Engineering,2016,024(016):47-49,53.