

# A continuous integration environment building based on the research of C#.NET

**Jintan Zhu<sup>1\*</sup>, Xiangfeng Liu<sup>2</sup>, Wei Wei<sup>3</sup>, Peiyi Shen<sup>4</sup>**

<sup>1</sup>*Xi'an Railway Vocational and Technical institute, Xi'an 710014, China*

<sup>2</sup>*Shaanxi Institute Of Technolog Shaanxi, Xi'an 710300, China*

<sup>3</sup>*School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China*

<sup>4</sup>*National school of Software, Xidian University, Xi'an 710071, China*

*Received 1 October 2014, www.cmmt.lv*

---

## Abstract

In the software development process of the past, integration is a very painful thing, usually a long time to do integration, as a result, it can cause many problems. Continuous integration, which is the core of the agile development practice, was integrated in one day more than ten times or even dozens of times, so often integration can minimize conflict. Due to the frequent integration, there are few changes to be made each time, even under integration failure conditions; it is easy to locate errors.

*Keywords:* continuous integration; positioning

---

## 1 Introduction

Continuous integration (continuous integration) is a software development practice (Aslin's context and spoken word recognition in a novel lexicon). After students learned C#.NET, had a certain understanding of software development. But for the team cooperation, they still felt very strange. So through continuous integration, the team members could frequently integrate their work between them. It is not a simple assembly software but it is the core of the software development process. Through constantly running tests, ensuring software's existing functions are not destroyed, automatically analyzing the states of the existing code (with and without repeated logic, the complexity of the code, etc.) and publishing the related reports (circle in the theory of agile development methods of science and technology inform) [1-12].

## 2 Domestic and foreign in-trend analysis

The acceptance time of a popular software development method by the industry may take ten years or twenty years. It has been a history of more than 20 years since the Scrum and XP were created and applied to industry (agile development of related technologies in the application of computer application and software). Lean software development mode has nearly 10 years of history. Now, these agile methods are achieving fine development, more and more people begin to pay close attention to them [8,13-18].

On the other hand, various types, various size companies have gradually started to focus on agile

development. Technology giants such as Google, Yahoo, IBM and Microsoft have used agile development for many years. They usually choose certain development team to try agile development mode, and then put the development experience of generalizability to other teams. Small and medium-sized software companies with flexible innovation characteristic, are more suitable for agile development. Many companies have already put the development team to fully transition to agile development mode. According to Forrester, "Dr. Dobb's Journal", and the Methods and Tools industry surveys of 2009 to 2011, a report in the United States and the European Union, understanding of agility and adoption is growing at about 50% per year. Obviously, agile methods will be at early majority phase to accelerate the adoption rate of growth, and will be more widely accepted by the enterprises (data integration environment database clustering algorithm based on similarity is computer engineering) [14-18].

Agile has been adopted in China. China's software development groups have started using agile methods in recent years, but they only account for very small portion of relevant organizations and companies. So we can say that agile adoption rate in China is still in the early stages of the technology adoption life cycle, trial (early adopters) phase.

## 3 Why do we need continuous integration?

In the software development process of the past, integration is a very painful thing, usually it takes a long time to do an integration, which can cause many problems, such as the build failings or failing unit test, usually it cost

---

\* *Corresponding author's* e-mail: zjt\_88@126.com

enormous human resources to locate and solve the problem. Integration of low frequency can be collected the code change feedback to developers, not to mention in the continuous incremental development can guarantee the stability of product functionality and reliability (Based on a continuous integration environment of software process management) [9-13].

Continuous integration at the core of the agile development practice, integrated in one day more than ten times or even dozens of times, so often integration can reduce conflict. Due to the integration very frequent, each integrated change has very few, even integration failure is easy to locate errors. Microsoft C#.NET on the Web(Continuous integration in the project analysis and research on computer programming skills and maintenance), WinForm application development, with the advantages of rapid and obviously, build based on C#.NET last will contribute to the stability of the product changes, and timely feedback. Summarized as follows:

1. The value of continuous integration;
2. Reduce risk, reduce manual process, generate the build results, and a sense of security;
3. The automation of continuous integration;
4. Automated testing, automatic generated can be deployment of the finished product, finished product will automatically deployed to the approximate conditions, automatic run regression tests, measurement reports automatically generated (announces upgrade to continuous integration and delivery server).

**4 How to deploy?**

Part of the agile practices in some organizations and companies have been carried out, but if adopted agile, is a huge task. Not only is the agile infrastructure, more important is the change of the whole team consciousness. This article mainly from the aspects of agile infrastructure, introduce how to build a continuous integration environment based on C#.NET, including access to the source code, compile the source code, perform all testing alarm, check the code and test coverage analysis, etc (Continuous Integration Server: Developers Can Now Test More, Configure Less).

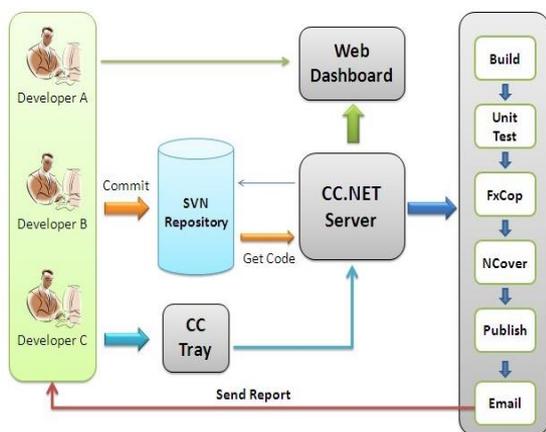


FIGURE 1 Continuous integration of the typical scenario

The main content includes:

1. Set up a continuous integration server;
2. Build a continuous integration environment;

Continuous integration of the typical scenario is as follows Figure 1.

**5 Set up and management of the continuous integration server**

Firstly, determine the installation server of CC.NET, located within the local area network (LAN). CC.NET of course ability is not limited to this, should be can extend the functionality of a lot of other (Scaling Agile Technical Practices: Implementing Continuous Integration to Enable Lean) [14-17].

CC.Net name "IBM Rational ClearCase Client for Visual Studio. NET". As is known to all, the ClearCase Thoughtworks is one of the flagship product of development of the company, is a leading global software configuration management tool, it is widely used in many enterprise software engineering practice, has many enterprise users. And Visual Studio 2008 (VS2008) is Microsoft's powerful software development integration environment, it is Microsoft. Net Framework is indispensable to develop software tools, Visual Studio. Net an important role in the world, also has many supporters. And CC.Net is ClearCase perfect integration in Visual Studio [11-15].

CC.Net name "IBM Rational ClearCase Client for Visual Studio. NET". As is known to all, the ClearCase Thoughtworks is one of the flagship product of development of the company, is a leading global software configuration management tool (Embodied concept formation and reasoning via neural-symbolic integration). it is widely used in many companies the ClearCase owning independent operating interface, the user can through the ClearCase Tray was such client software to operate and control, and it also provides several ways, there are Abort the Build, Stop the Project and Display a Web Page, Copy the Build Label, which use the Display a Web Page can be intuitive view of integration developer compiled code of the Page after the various data in the current Project(Time to accelerate integration of human factors and ergonomics in patient safety).

CC.Net name "IBM Rational ClearCase Client for Visual Studio. NET". As is known to all, the ClearCase ThoughtWorks is one of the flagship product of development of the company. It is a leading global software configuration management tool, it is widely used in many companies claim to the continuous integration server set up and functional analysis, the function module on the combination of the corresponding debugging, to ensure the correct operation of all functions and display(Study on Ship Design Information Integration Model).

5.1 ACCESS TO THE SOURCE CODE

The source code is the core of the company's products, is one of the company's wealth. Different version at the time of development, the source code in the previous version on the basis of constantly accumulating. Every time for continuous integration, integration needs to get the latest source code and guarantee the integration.

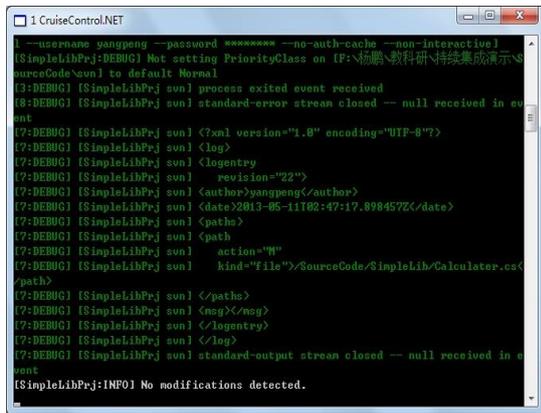


FIGURE 2 Authentication structure

5.2 COMPILE THE SOURCE CODE

After the source code is compiled into an executable program can be run. After obtaining the source code, compile the source code can be found that the code is flawed. Of course to meet Microsoft's compilation rules and norms, thus avoid the process of compilation errors, for subsequent code detection and operation obstacles [18].

5.3 CODE THE ALARM

By the compiler or the alarm check tool to analyze the source code is not in conformity with the specifications, timely discover and modify, to enhance the quality of source code:

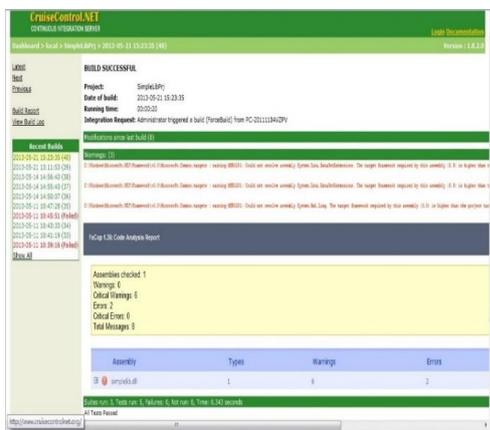


FIGURE 3 Code the alarm

5.4 UNIT TESTING

Through the test cases cover, timely feedback to the submitted code can guarantee the stability of the original

function, or commit the code in the original function and the effect of early find and solve the problem, avoid costly human when product integration test to solve the defects problem. Nunit execution by Nunit(Untangling the intracellular signalling network in cancer — A strategy for data integration in acute myeloid leukaemia). Write test cases and use case and the output. Already can through the console command to drive the case execution. Above is the result of performing use-case GUI way: use case successes, failures and errors may be collected data. In the validation process, through the console command execution results can be saved as XML files, then using XSL to report shows.

5.5 CODE COVERAGE

By performing testing, test results, and source code analysis, check the code and which are not covered. No covered code, either scenario is missing, either there is something wrong with the code processing. May be testing to ensure function as much as possible. Using Ncover, Nunit execution results for analysis, to check the coverage of the test function. Main branch from the code coverage measurement. The higher coverage, function test, the more fully, and the smaller risk of errors(The integration of home-automation and IPTV system and services).

5.6 CHECK CODE COMPLEXITY

The more concise code easier to read, the more easy to maintain. Through checking the complexity, can in time found that excessive complexity of code or function, and then constantly refactoring, code structure adjustment, ensure the quality of the code. Use FxCop tool for c # coding specification, design, etc., use FxCop rules for inspection(A multi-layered approach to CRM implementation: An integration perspective). Increase code quality through this check can class design and quality.

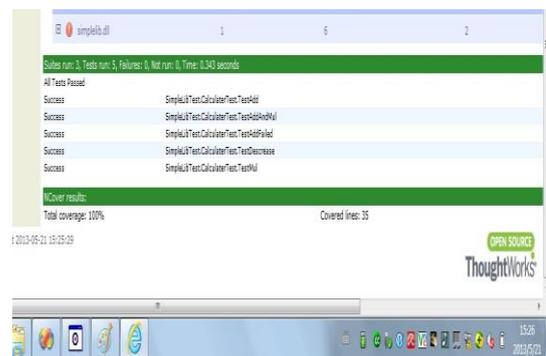


FIGURE 4 Check code complexity

6 Conclusion

Finding out the most suitable C#.NET development tools, integrating it into a continuous integration environment can be realized through the analysis of the existing popular continuous integration tool (Supply chain integration and

service oriented transformation: Evidence from Chinese equipment manufacturers). According to the need of the build steps, development support tools and reporting transformation tools, will form a complete system of continuous integration environment, they will also contribute to enterprise application integration environment, promoting enterprise agility practice capability.

In terms of agility development, the popularity of a development method is a process of accumulation. A good development method is established through constant practicing and validating and it must be effective (An analysis and strategies for implementation to meet mobile commerce challenges). There is no clear dividing line that marks the degree and level of the popularity of the agility development, at least for now, agility is still under development. More projects are using agility in practice. the spread of the agility concept and successful agility

cases are growing. It is stressed that agility is not a "silver bullet", in light of there is no universal development mode, only constant practicing will create better development.

### Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments. This program is supported by Scientific Research Program Funded by Shaanxi Provincial Education Department (Program No.2013JK1139) and by China Postdoctoral Science Foundation (No.2013M542370) and by the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20136118120010). This work was partly also supported by the National Natural Science Foundation of China (No.61072105& 61272283& 11301414& 11226173).

### References

- [1] Reville K P, Tanenhaus M K, Aslin R N 2008 Context and spoken word recognition in a novel lexicon *Journal of experimental psychology. Learning, memory, and cognition* **34**(5) 1207-23
- [2] Liu S 2011 Circle in the theory of agile development methods of science and technology inform 2011 07
- [3] Chen N, Chen W 2011 Agile development of related technologies in the application of computer application and software 2011 04.
- [4] Zheng K, Liang Z, Zheng W 2011 Data integration environment database clustering algorithm based on similarity is computer engineering 2011 10
- [5] Wu Z 2011 Based on a continuous integration environment of software process management Fujian 2011 06.
- [6] Chen G, Qiang B B 2011 Continuous integration in the project analysis and research on computer programming skills and maintenance 2011 12
- [7] Atlassian announces upgrade to continuous integration and delivery server *Telecomworldwire* (M2) 2012
- [8] Atlassian Launches Bamboo 4 Continuous Integration Server: Developers Can Now Test More Configure Less *Business Wire* (English) 2012.
- [9] Research and Markets: Scaling Agile Technical Practices: Implementing Continuous Integration to Enable Lean Business Wire (English) 2012
- [10] Gurses AP 2011 Time to accelerate integration of human factors and ergonomics in patient safety *BMJ quality & safety* **21**(4)
- [11] Min J 2010 Embodied concept formation and reasoning via neural-symbolic integration *Neurocomputing* **74**(1)
- [12] Yao J-Z 2011 Study on Ship Design Information Integration Model *Energy Procedia* 2011 11
- [13] Jørgensen K M 2010 Untangling the intracellular signalling network in cancer – A strategy for data integration in acute myeloid leukaemia *Journal of Proteomics* **74**(3)
- [14] Umberger M 2008 The integration of home-automation and IPTV system and services *Computer Standards & Interfaces* **31**(4)
- [15] Finnegan D J 2009 A multi-layered approach to CRM implementation: An integration perspective *European Management Journal* **28**(2)
- [16] He Y Q 2011 Supply chain integration and service oriented transformation: Evidence from Chinese equipment manufacturers *International Journal of Production Economics* **135**(2)
- [17] Swallows D 2007 XML and WML integration: An analysis and strategies for implementation to meet mobile commerce challenges *Computer Standards & Interfaces* **29**(1)

Authors	
	<p><b>Jintan Zhu, May 1981, Weinan, Shaanxi, China.</b></p> <p><b>Current position, grades:</b> professional teacher, lecturer.  <b>University studies:</b> master's degree majoring in Computer Science and Technology at Xi'an Electronics and Science &amp; Technology University.  <b>Scientific interests:</b> software engineering.  <b>Publications:</b> 4.</p>
	<p><b>Xiangfeng Liu, September 1982, Huxian, Shaanxi, China.</b></p> <p><b>Current position, grades:</b> professional teacher, lecturer.  <b>University studies:</b> Computer Science and Technology at Lanzhou University of Technology.  <b>Scientific interests:</b> computer network technology and information construction.  <b>Publications:</b> 1 book.</p>
	<p><b>Wei Wei, China.</b></p> <p><b>Current position, grades:</b> assistant professor at Xi'an University of Technology.  <b>University studies:</b> MS and PhD degrees at Xi'an Jiaotong University in 2005 and 2011, respectively.  <b>Scientific interests:</b> wireless networks and wireless sensor networks application, mobile computing, distributed computing, and pervasive computing.</p>
	<p><b>Peiyi Shen, China.</b></p> <p><b>Current position, grades:</b> professor at the National School of Software at XiDian University. Member of IEEE.  <b>University studies:</b> PhD in XiDian University in 1999.  <b>Scientific interest:</b> computer vision, volume visualization and ITS based WSN applications.</p>