

Open Source Software Developments in XP Style

Position Paper for the 1st Workshop on Open Source Software Engineering

Yoshiyuki NISHINAKA

SRA Key Technology Laboratory, Inc.

3-12, Yotsuya, Shinjuku-ku, Tokyo 160-0004, Japan

<mailto:nisinaka@sra.co.jp>

<http://www.sra.co.jp/people/nisinaka>

I've been working on three open source softwares for the last 10 years:

- XME (eXtensible Meta Environment): 1989/07 - 1992/03
XLISP and X Window System based GUI applications development environment.
- Tcl/Tk Japanization Kit: 1992/09 - 1996/11
A patch to handle Japanese character sets with Tcl/Tk.
- Jun: 1997/11 - now
A class library to create 3D graphic multimedia applications for both Smalltalk and Java.

Jun is our original open source software. There is a group with some members for the project inside our company. Jun is distributed to the world via Internet and the users give us feedbacks. Some of the users who gave us an important feedback are also copyleft (Jun is distributed under GNU GPL) holders of Jun. In this case, I am one of the core members of the software.

Tcl/Tk is a script language and its toolkit. It is originally developed by Dr. John Ousterhout and is distributed as open source software. The original Tcl/Tk cannot handle Japanese character sets (and other multi-byte character sets). I modified the source code, made a patch to handle Japanese character sets, and distributed it. Sometimes we found bugs in the original source code and gave feedbacks to Dr. Ousterhout, and sometimes they asked us some questions and comments for internationalization of Tcl/Tk. In this case, I was a contributor to Tcl/Tk.

The XME case is a mixture of the above two. XLISP and X Window System are both open source softwares. We made our original command sets and libraries as a development environment, and at the same time, we made some modifications to XLISP and X Window System. We distributed XME as open source software and contributed to XLISP and X Window System.

From the experiences of these projects, I learned how to maintain source files, how to cooperate with physically distributed project members, how to distribute open source softwares, and how to communicate with contributors.

We did not pay much attention to methodology and process to carry out the projects. Looking back to the past days, I realize that the projects were preceded with XP (extreme Programming) style in someway.

XP is not such a new methodology. As Kent Beck says in "Extreme Programming Explained", most of the practices in XP are as old as programming. In our projects, we didn't have any specific customers and any deadlines to release, so some of the practices, such as "The Planning Game" or "On-site Customer" were not applicable. Most of the others, such as "Small Releases", "Simple Design", "Testing", "Refactoring", "Collective Ownership", and "Coding Standard" are spontaneously applied.

In the case of Jun project, we first develop new features with Smalltalk version of Jun. Then port them into Java version. Actually, it is not just a porting. We can say we design Jun with Smalltalk and implement it with Java.

All of the Jun project members have their responsible part. At the same time, all of the Jun project members care about the entire Jun. If a user give us a feedback and say "this feature doesn't work on my Linux box", almost all of the members who can use Jun on Linux may try the user's problem on their own environment and report the results to our project mailing list.

We write test cases first. Test cases are provided as class methods (in Smalltalk version, or static methods in Java version) and they are part of the Jun system. So the users can see the test cases and understand how to use the features related to the test cases.

We never tried our project in XP style. The word "XP" did not exist when we start the project. We preceded our project with XP style in consequence. XP can be a good methodology for open source software projects if:

- it is not a big project
- there is one or more core members
- there are some contributors
- a project is not for money :-)
- a project is not for any specific customers :-)