An Example of Product Refactoring

Jan Mertens
Alcatel R&I, Telecom Services Project
F.Wellesplein 1, 2018 Antwerpen, Belgium
+32 (0) 3 240 81 61, Jan.Mertens@alcatel.be

Linas Maknavicius
Alcatel R&I, Telecom Services Project
Route de Nozay, 91460 Marcoussis, France
+33 1 69 63 44 51, Linas.Maknavicius@alcatel.fr

ABSTRACT
This report describes the start-up multi-site process of industrializing a product using XP practices. The fact that the development group was distributed across two countries added some extra challenges to the process.

Keywords
Multi-site, industrializing a service platform

1 INTRODUCTION

The Research & Innovation (R&I) division at Alcatel is a breeding pool of new ideas. The ultimate goal of the Telecom Services project is to supply an operator with an application service enabling platform placed on top of network elements. The industrialization activity (refactoring an existing prototype of the platform) is carried out by two development teams each consisting of a dozen people, in Antwerpen, Belgium and Marcoussis, France. The rest of this report will concentrate on our experience with applying XP practices.

2 ITERATION 0

First, we started with a short three week iteration 0. During one week we defined the stories. In the next two weeks both teams were split into two groups. One small group per site implemented the stories and the other one defined the stories for the next iteration. During the story definition phase we used the CRC-card approach. The defined stories were put in a simple electronic database with a web front-end. In this way people at both sites could easily add notes to the stories and report planned and actual days taken for the implementation of the story.

At the end of the iteration 0 the implementation group gave a successful demo to the rest of the team and the marketing people.

3 ITERATION 1

At the first day of iteration 1 we organized at each site an iteration kick-off meeting. In this session the new stories defined in iteration 0 were discussed and the cost in ideal days was estimated. The integration of the stories from both sites happened in the last days of the iteration.

4 LESSONS LEARNED

Communication across sites

Weekly status reports were sent to the other site after a series of stand-up meetings. Most cross-site communication happened via email, ICQ messages and by updating the story database with status and notes. We feel however that this is far from being sufficient and that oral communication could usefully be complemented by written assessment in order to share the knowledge.

What we need is a signal that a story is finished and integrated in the code base. On top of this, we need tests which prove that the story is implemented and that can act as the story documentation.

Configuration Management & Integration

During iteration 0 all team members worked on a single branch in the configuration management system. In iteration 1 we experimented with a development stream in each site and a common integration branch. This concept is easy to install but it doesn't allow clean story deliveries. For this reason we are planning to introduce a stream per story and an integration branch per site.

Pair Programming

The concept of pair programming has proven to be very powerful. We saw the practice growing in the teams.

Functional Testing

During our iterations we experienced the importance of more global functional/system tests. However we haven't yet found the best approach. For functionality implemented in Java there is no big problem. But on the border of our system we have functionality encapsulated in browsers or configurations in network elements. It is not easy to get here full control for enabling continuous integration and testing mechanisms like described in [1].

5 CONCLUSION

Contrary to the common belief that XP for multi-site is only advisable for your competitors [2] we would like to prove that applying XP practices to a distributed team can lead to a better product. On the one hand two teams need more communication (which means significant overhead) but on the other hand fast ramp-ups are possible and cultural differences can be complementary and synergetic if managed properly.

REFERENCES

1. M.Fowler. Continuous Integration. On-line at http://www.martinfowler.com/articles/continuousIntegration.html
2. Mailing list at http://groups.yahoo.com/group/extremeprogramming/message/23608/ .