

Tools in a Development Process

Gertjan Zwartjes^{*+}, Joost van Geffen^{*+}, and Bruce Watson⁺
 {gzwartjes, jvgeffen}@research.intersoft.nl, bruce@bruce-watson.com

*Intersoft Software Research / +Espresso
<http://espresso.cs.up.ac.za>

Software Development Tools

It is inevitable that people will make mistakes in the development of software. These errors often cost a lot of time and resources to fix and are serious obstacles in the process of development. Tools can decrease the development effort by assisting in managing mistakes in three ways:

1. By preventing errors
2. By detecting and locating errors
3. By fixing errors

“Tools amplify your talent. The better your tools, and the better you know how to use them, the more productive you can be.” [Hunt and Thomas, 1999].

A Case Study

Automation is one such way to assist in preventing, detecting, locating, and fixing errors. A study of two automated build systems showed that the tool decreased the build time of the system to approximately 5% of the original build time. An automated deployment system was investigated in one of the studies as well, and showed a reduced deployment time of approximately 3% of the original deployment time.

Tool Selection with a Tool Matrix

Because well-chosen and appropriately used tools can contribute greatly to the success of a

project, the process of tool selection is very important. The tool matrix [Zwartjes *et al.*, 2005] is a visualization of the tool evaluation and selection process. The columns of the matrix contain the required properties of tools — grouped into categories — and the rows of the matrix list the tools and their compliance to those properties. As part of defining the need for a specific requirement, a criticality is assigned to each of the required properties. These criticalities are used as criteria in choosing the final set of tools.

	General		Category α								Category β								
	Property 1	Property 2	Property 3	Property 4	...	Property 8	Property 9	Property 10	Property 11	Property 12	...	Property 21	Property 22	Property 23	Property 24	Property 25	
Criticality	H	H	M	M	M	M	H	H	H	H	H	H	L	H	M	M	M	M	M
Tool A																			
Tool B																			
Tool C																			
Tool D																			
Tool E																			
Tool F																			
Tool G																			
Tool H																			
...																			

An example of a tool matrix, listing imaginary tools and properties. For the criticality an indication of high (H), medium (M), and low (L) is chosen.

References

[Hunt and Thomas, 1999] Andrew Hunt and David Thomas. *The Pragmatic Programmer: From Journeyman to Master*. Addison-Wesley, 1999.

[van Geffen and Zwartjes, 2005] Joost van Geffen and Gertjan Zwartjes. An agile approach supported by a tool environment for the development of software components. Master's thesis, Technische Universiteit Eindhoven, January 2005.

[Zwartjes *et al.*, 2005] Gertjan Zwartjes, Joost van Geffen, Derrick Kourie, Andrew Boake, and Bruce Watson. Industry experience in using an abstract model to select software development tools. *To be published in SAIEE Special Issue on Software Engineering*, 2005.