

## **CVS integration with notification and ch**



that the developers actually sent the email before they checked-in code to give “a brief description of the impact that their work [changes] will have on other’s work”. The purpose of this was to give others time to “prepare for and reflect about the effect of their changes”, often resulting in people coming to ask about the change or asking for a delay, etc.

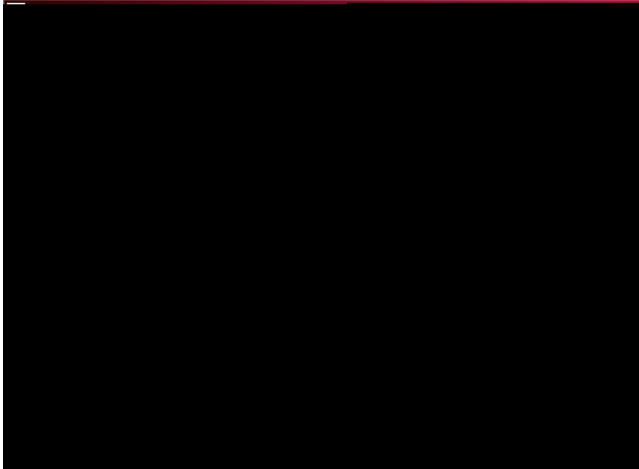
Yamauchi et al. [23] also studied two different distributed open source projects, both using CVS, where mailing lists were pivotal for coordination and awareness. Before check in, developers would extract “the difference between the modified version and the central master code with [the command] ‘diff’ and then submit the differences to a mailing list” (p333).

In all of these cases email serves to meet an informal communication and awareness need, but the informal communication takes place in parallel to, but separated from the CVS logs that anchor the discussions. They also require explicit effort (apart from the automated sending of changes) on the part of the developers to send the initial message and further describe or discuss the changes.

Despite this, most developers reported that they felt that the combination of a CM tool and email worked well. Email provided a way of making a change to the code base without the need to

making a change to the code base without the need to

registered interest. Because the underlying event model in Elvin is very generic, a producer or a consumer can be anything from a software component to a person and it can be put to uses ranging from systems-oriented middleware messaging, to people-oriented filtering of information feeds from source



**Figure 2. tickertape conversation triggered by CVS commit message.**

In the example presented in figure 2, it is late at night and Phelps is working from home fixing a bug in some code. When he is finished, he checks the file back into the CVS repository, entering the comment “The gap doesn’t actually need to draw anything”. This check-in event causes an Elvin notification to be generated stating the name of the modified file, ‘gap.c’, and the associated comment. The notification is then sent by the server to people subscribed to the ‘elvin’ group. David (‘d’) is working late back in the office and sees that Phelps has made some changes to the ‘gap.c’ file (history line 1). He sends a message joking “and you tell me to go to bed!!!!;-)” (line 2). Phelps and David then engage in some light-hearted banter about their working habits (lines 3-4). Phelps goes on to explain a bit more about what else he has been working on and they have a short discussion around that work (lines 5-7). In the middle of the discussion, David has a problem with his tickertape.

Tj 37 0 0 37 336 8 Tm(t) Tj 37 0 0 37 768 0 Tm ( ) Tj 37 0 0 37 360 0 Tm (h) Tj 37 0 0 37 37 01Tm (l) Tj 37 0 0 37 388 0 Tm (l) Tj 3



**Table 2: Frequency of occurrence of types of messages**

Category of log entry	Frequency of messages coded as containing text belonging to the category (percentage of total)	
	Pre Elvin notifications	Post Elvin notifications
<b>Description</b>	168 (58.1)	112 (61.9)
<b>Basic description</b>	52 (17.9)	20 (11.0)*
<b>Effect</b>	68 (23.5)	49 (27.1)
<b>Rationale</b>	55 (19.0)	40 (22.1)
<b>Future/ incomplete</b>	14 (4.8)	8 (4.3)
<b>Value judgement</b>	12 (4.2)	5 (2.8)
<b>Empty message</b>	18 (6.2)	0 (0)**
<b>Invitation</b>	0 (0)	0 (0)
<b>Landmark</b>	2 (0.7)	3 (1.7)
<b>Unsure/ hopeful</b>	1 (0.3)	2 (1.1)
<b>Communication</b>	2 (0.7)	1 (0.6)
<b>Named other developer</b>	12 (4.2)	7 (3.9)
<b>Smiley</b>	1 (0.3)	1 (0.6)

*Pre Elvin notifications: N=289; post Elvin notifications: N=181.*

*Pre and post frequencies were compared for each of the categories using the  $\chi^2$  statistic.*

*\* $p < 0.05$*

*\*\* $p < 0.01$*

As table 2 shows, there was a decrease in the number of basic descriptions and empty log messages after CVS messages started to be sent as Elvin notifications. There was no significant change

when the cvs2ticker script was implemented, were deleted. 12129 log entries were used in the analysis

### **6.3 Word Counts**

Word counts were calculated for each of the remaining messages in the Elvin CVS log. Counts ranged from 0 words to 119. The mean word count was 9.0 (standard deviation = 8.3)

The correlation of word count with time was calculated to investigate whether the increase in commit message length found in the Orbit log might extend to the much larger Elvin CVS log. A small, highly significant positive correlation was found between word count and time ( $r=.071$ ,  $p<.001$ ), a finding in line with the interpretation that the increased public availability of CVS commit messages afforded by tickertape may have encouraged developers to write longer descriptions of code changes.

Correlations of commit message word counts with time for individual developers are tabulated in table 3.

**Table 3: Correlations of word count with time for individuals**

**Develope**



The volume of tickertape

- [4] Curtis, B., Krasner, H., and Iscoe, N. 1988. A field study of the software design process for large systems. *Commun. ACM* 31, 11 (Nov. 1988), 1268-1287. \_
- [5] De Marco, T. and Lister, T. *Peopleware: Productive Projects and Teams*. 2<sup>nd</sup> Ed. Dorset House Publishing, New York, 1999.
- [6] de Souza, C.R.B., Redmiles, D., Dourish, P., 'Breaking the Code', Private and Public Work in Collaborative Software Development. In Proceedings of the *International Conference on Supporting Group Work* (Group 2003) (Sanibel Island, FL, November 2003) 105-114.
- [7] Fitzpatrick, G., Kaplan, S., Mansfield, T., David, A., and Segall, B. 2002. Supporting Public Availability and Accessibility with Elvin: Experiences and Reflections. *Comput. Supported Coop. Work* 11, 3 (Nov. 2002), 447-474.
- [8] Fi