Status Report on Software Configuration Management Impact Study

Alexander L. Wolf
Department of Computer Science
University of Colorado at Boulder
Team

◆ Lead Authors
  – Jacky Estublier, U. Grenoble/IMAG (Adele)
  – David Leblang, retired (Clear Case, DSEE)

◆ Contributing Members
  – G. Clemm, Rational (Clear Case, Odin)
  – R. Conradi, U. Trondheim (EPOS)
  – A. van der Hoek, UCI (NUCM)
  – W. Tichy, U. Karlsruhe (RCS)
  – D. Wiborg-Weber, Continuus (CCM)
Domain of SCM

- Managing a repository of components
  - Version Control; Product Models; Composition and Selection
- Helping engineers in their usual activities
  - Building (derived object control); Work Space Control
- Controlling and supporting the process
  - Change Control; Cooperative Work; Process Support
Growing Market for SCM Products

◆ Ovum
  – 25% mainframe; 15%-20% workstations; 5%-10% PC

◆ Gartner
General Plan of Study

- Examine characteristics/features of leading products in SCM market
- Assume that products used in practice
- Trace characteristics/features back to research ideas and prototypes
- Try to establish arguments for/against influence of research on practice (via products)
An Argument: Research/Product Timing

- Research initiative was shared between academia and industry
- Some research tools were seriously used in practice
  - Make, RCS, Odin, Adele ...

from A. van der Hoek
An Argument: Professional Interaction

- Product architects present at nearly all SCM workshops (1988-2001)
  - Cagan, Clemm, Dart, Leblang, Wiborg-Weber, ...
An Argument: Testimonials

◆ Initial research perspective
  – “We invented almost everything …”
  – “Tools are only an engineering issue …”

◆ Initial vendor perspective
  – “Research had very little influence …”
  – “We do not sell ideas, but tools. We (re)invented everything we needed…”

◆ After some discussion, a much more balanced perspective emerged from both communities
Some Lessons Learned (1)

Vendors tend to consider that impact is restricted to...

- *algorithms* (e.g., differencing)
- *pieces of reusable code* (e.g., RCS)

and not...

- *concepts* (e.g., hierarchical workspaces)
- *architectures* (peer-to-peer repositories)

which are often seen as “engineering common sense”
Some Lessons Learned (2)

- Researchers tend to consider that... 
  
  *precedence*
  
  *concepts*
  
  *prototypes*

  are sufficient as impact and ignore...
  
  *efficiency*
  
  *usability*
  
  *reliability*

  dismissing them as “engineering common sense”
Some Lessons Learned (3)

- Both are right, both are wrong
- A good idea is had more than once
- Vendors have disincentives for distributing credit for ideas
- Researchers have incentives for claiming credit for ideas
- Research and productization both require engineered creativity
Conclusion

- SCM is a successful field
- Research provided many inputs and was clearly influential
- Vendors successful in finding/adapting ideas to fit customer needs
- Many ideas tried by researchers have not (yet) found their way into products/practice
- Interplay between vendors and researchers exists, but not any easy relationship