



***“An Empirical Characterization
of Scientific Software
Development Projects According
to the Boehm and Turner Model:
a Progress Report”***

Carlton A. Crabtree, A. Gunes Koru*, Carolyn Seaman
UMBC, Department of Information Systems
Baltimore, MD, USA
cac1, gkoru,cseaman@umbc.edu

Hakan Erdogmus
National Research Council of Canada (NRC/CNRC)
Hakan.Erdogmus@nrc-cnrc.gc.ca

Research Motivation

- **Support awareness**
 - Software engineering principles in the scientific software development community
- **Studies suggest**
 - Agile methods are well suited to scientific software development
 - requirements emergence, iterative workflow, tacit knowledge in the scientific research process
 - These findings may not be representative of all factors
 - documentation needs, employee turnaround, non-experts, student developers, well defined and stable project requirements
- **An empirical approach**
 - Enhances our ability to understand the scientific project context without bias
 - Boehm and Turner Model provides a mechanism to categorize project characteristics





Research Goals

- ***To generate*** a set of testable hypotheses for future evaluation
- ***To study*** scientific software development project characteristics
- ***To increase our understanding*** of scientific software development project characteristics
- ***To provide guidance*** to scientific software developers
- ***To foster collaboration*** and discussion about qualitative research approaches at SECSE09

Related Work

- **CS&E Case Study Foundations:**
 - Software Development Environments, Carver et al.
 - High Performance Computing (HPC), Basili et al.
 - Risk management Analysis, Sanders and Kelly
 - Workflow Systems, Woollard et. al.
 - Scientific Software Development Models, Segal et al.
 - Interpretive analysis of agile methods using the Boehm and Turner model, Segal
- ✓ **Agile Studies:**
 - “*Exploring XP for Scientific Research*”, IEEE Software 2003, Wood et al.
 - “*Scientific Software Development at a Research Facility*”, IEEE Software, 2008, Ackroyd et al.
 - “*Agile methods in biomedical software development: a multi-site experience report*”, BMC Bioinformatics 2006, Kane et al.



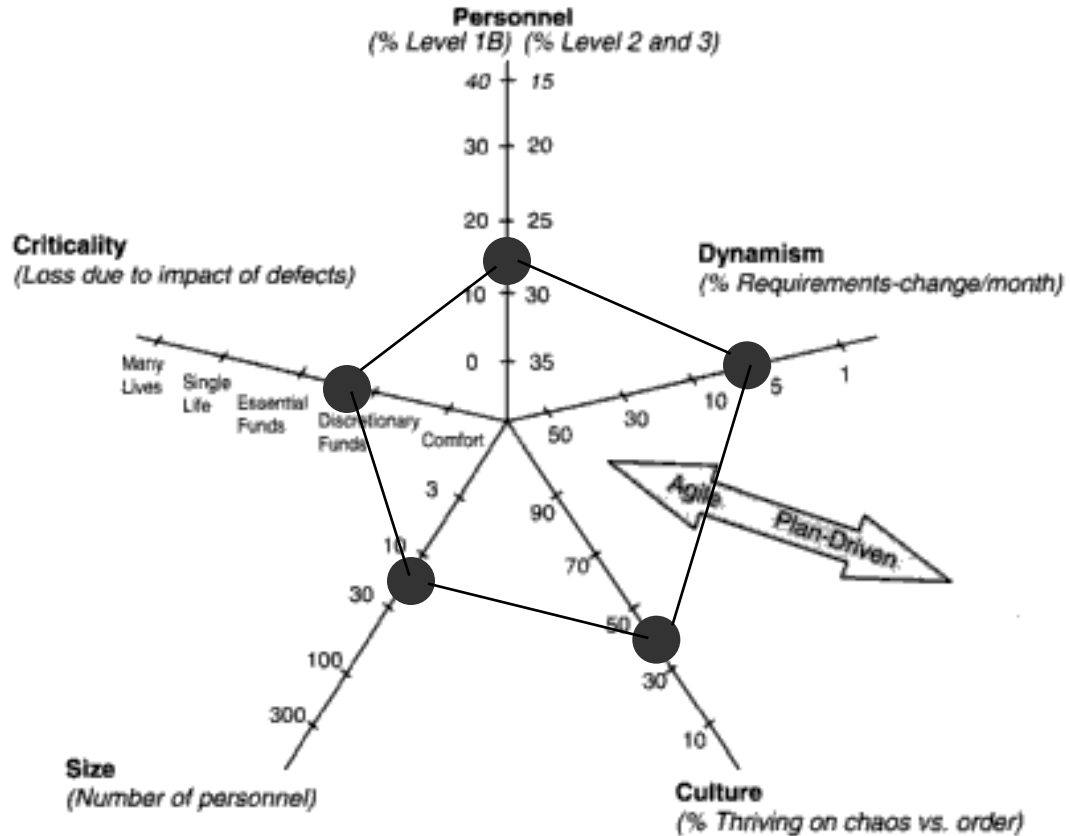
Findings from Literature Review

- Few studies that evaluated projects of various size and in multiple CS&E disciplines
- No identified studies that categorize project characteristics using the Boehm and Turner Model
- Concerns over participant exposure and wide variety of interpretation of software engineering methodology
- The Boehm and Turner model provides an approach to evaluate scientific software development projects not based on practices but actual processes

Boehm and Turner Model

- Provides criteria for comparing and contrasting project characteristics:
 - *Plan Driven Approaches*
 - Predictable and stable
 - Documented requirements
 - Systematic engineering processes
 - Requirements/Design/Build Paradigm
 - *Agile Methods*
 - High change environments
 - Undocumented requirements
 - Rapid responsiveness
 - Tacit Knowledge/Simple Design
- Characteristics
 - Application, Management, Technical, Personnel
- Critical Agility/Plan Driven Factors
 - Size, Criticality, Dynamism, Personnel, Culture

Dimensions Affecting Method Selection



[4] Boehm, B. and R. Turner, *Balancing Agility and Discipline: A Guide for the Perplexed*, Addison-Wesley, Boston, MA, 2003, page 56.

Data Collection

- Convenience sample
 - Government, academic and commercial scientific software development projects
- Projects of varying size
 - Number of software development personnel
- Interviews with scientific software developers:
 - Field Memos and Notes
 - Semi structured interviews (open and closed end questions)
 - Interview protocol tailored to elicit project characteristics without using software engineering “buzzwords”
 - Questions based upon our interpretation of the Boehm and Turner sub categorizations of agile and plan driven criteria

Planned Data Analysis

- *Stage 1:* Grounded theory method
 - Largely a creative process
 - Seeking patterns and trends in the data
 - Textual encoding of interview questions
 - Analysis and comparison of observations from field memos
- *Stage 2:* Reconciliation of interview data to the Boehm and Turner Model to be used as a baseline for analysis
- *Stage 3:* Produce polar chart representations of the data per scientific software development project

Limitations and Risks

- Our approach does not fully compare / contrast the data collection and analysis strategy with other studies
- Little prior experience with the empirical validation using this model in the CS&E domain (limited knowledge base)
- Potential difficulty in producing the representations consistent with the Boehm and Turner model
- Problems in interpreting participant responses based upon interview questions
- Our approach is somewhat dependent upon Boehm and Turner's model



Contributions

- Provides a greater understanding of scientific software project characteristics through an exploratory in depth qualitative analysis
- To our knowledge this is a new approach comprehensive approach for understanding CS&E project characteristics
- Creates a set of hypotheses for further evaluation and testing in different project environments
- Increases scientific software developer decision making ability about what types of processes might be appropriate
- Fosters additional discussion and collaboration

References

- [1] K.S. Ackroyd, S. H. Kinder, G. R. Mant, M.C. Miller, C.A. Ramsdale and P.C. Stephenson, “Scientific Software Development at a Research Facility”, IEEE Softw. 25, 4 (Jul. 2008), 44-51.
- [2] V. R Basili., J. C. Carver, D. Cruzes, L. M. Hochstein, J. K. Hollingsworth, F. Shull, and M. V. Zelkowitz, “Understanding the High-Performance-Computing Community: A Software Engineer's Perspective”, IEEE Softw. 25, 4 (Jul. 2008), 29-36.
- [3] Beck, K., Extreme Programming Explained: Embrace Change, Addison-Wesley, Boston, MA, 2000.
- [4] Boehm, B. and R. Turner, Balancing Agility and Discipline: A Guide for the Perplexed, Addison-Wesley, Boston, MA, 2003.
- [5] C. Coleman, R. O’Conner, “Using grounded theory to understand software process improvement: A study of Irish software product companies”, Information and Software Technology, 49 (2007) 654-667.



information
systems

- [6] D.W. Kane, M.M. Hohman, E. G. Cerami, M. W. McCormick, K.F.Kuhlman, J. A. Byrd, “Agile methods in biomedical software development: a multi-site experience report”, BMC Bioinformatics, (May 2006).
- [7] R. Sanders, D. Kelly, 2008, “Dealing with Risk in Scientific Software Development”, IEEE Softw. 25, 4 (Jul. 2008), 21-28.
- [8] C. B. Seaman. “Qualitative Methods in Empirical Studies of Software Engineering”, IEEE Trans. 25 (4) (Jul. 1999), 557-5723.
- [9] J. Segal, C. Morris, “Developing Scientific Software”, IEEE Softw. 25, 4 (Jul. 2008), 18-20.
- [10] J. Segal, “When Engineers Met Research Scientists: A Case Study”. Empirical Softw. Engg. 10, 4 (Oct. 2005), 517-536.
- [11] J. Segal, “Models of Scientific Software Development,” Proc. 2008 Workshop Software Eng. in Computational Science and Eng, (Secse 2008).
- [12] Strauss, A. and J. Corbin, Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, NY: Sage, New York, NY, 1998
- [13] W.A. Wood and W. L. Kleb, “Exploring XP for Scientific Research”, IEEE Softw. 20, 3 (May. 2003), 30-36.



information
systems

UMBC

AN HONORS
UNIVERSITY
IN MARYLAND