

# Special Issue of IEEE TSE on MSR!!

---

- Authors of selected workshop papers will be invited
- Everyone encouraged to submit through an open Call for Papers
- Reviewers will be contacted, please help out!
- Editors:
  - Ahmed Hassan, Ric Holt, Audris Mockus, Philip Johnson
- Deadlines:
  - Intent to Submit: Oct 1, 2004
  - Submission Deadline: Oct 22, 2004
  - Notification of Acceptance: April 18, 2005
  - Publication Date: October 2005



# Common Themes & Future Directions

# A New Opportunity: Lots of software repositories ...

---

- Until recently, a few archives on software development
- Now, OSS gives us a corpus of software archives
- Easier to predict bugs
- Easier to validate utility of software metrics
- Conclusive studies of software evolution
- Predictions about software easier to validate
- What do developers do: process as it occurs
- Etc etc etc

# Six Sessions at MSR-04

---

1. Infrastructure & extraction
2. Integration & presentation
3. System understanding & change patterns
4. Defect analysis
5. Process & community analysis
6. Software reuse

# 1. Infrastructure & extraction

---

- MSR Matures?
  - Std extractors?
  - Std guinea pigs?
  - Std benchmarks?
  - Std statistical methods?
  - MSR conference?
  - Funding for MSR?

## 2. Integration & presentation

---

- Data integration: Combining data from multiple sources?
- What sources of data beyond code?
- OSS data not as detailed as industrial data
- Presenting results:
  - Charts & graphs?
  - Tables?

# 3. System understanding & change patterns

---

- Software evolution: lots of new data. How does change occur?
- Clone analysis
- Co-change predicts co-change. How well? Useful info?
- What do developers do? Will the repository tell us?

# 4. Defect analysis

---

- Bugs & change predict bugs. How well & in what ways? Precision & recall.
- Ranking risk. By file? By function? Early warning?
- “Complexity” predicts bugs, or does it?
- Where do bugs come from & where do they go



# 5. Process & community analysis

---

- Does repository tell us about what developers do? Social networks & collaboration: What do they do together? Are some programmers very much better or worse? Does experience help?
- Ethics: Is big brother watching?

# 6. Software reuse

---

- Googling to find & reuse software? Time to use info retrieval techniques on software?
- Repositories can tell us how to re-use software, or can they?

# A New Opportunity: What next??

---

- MSR 2005?