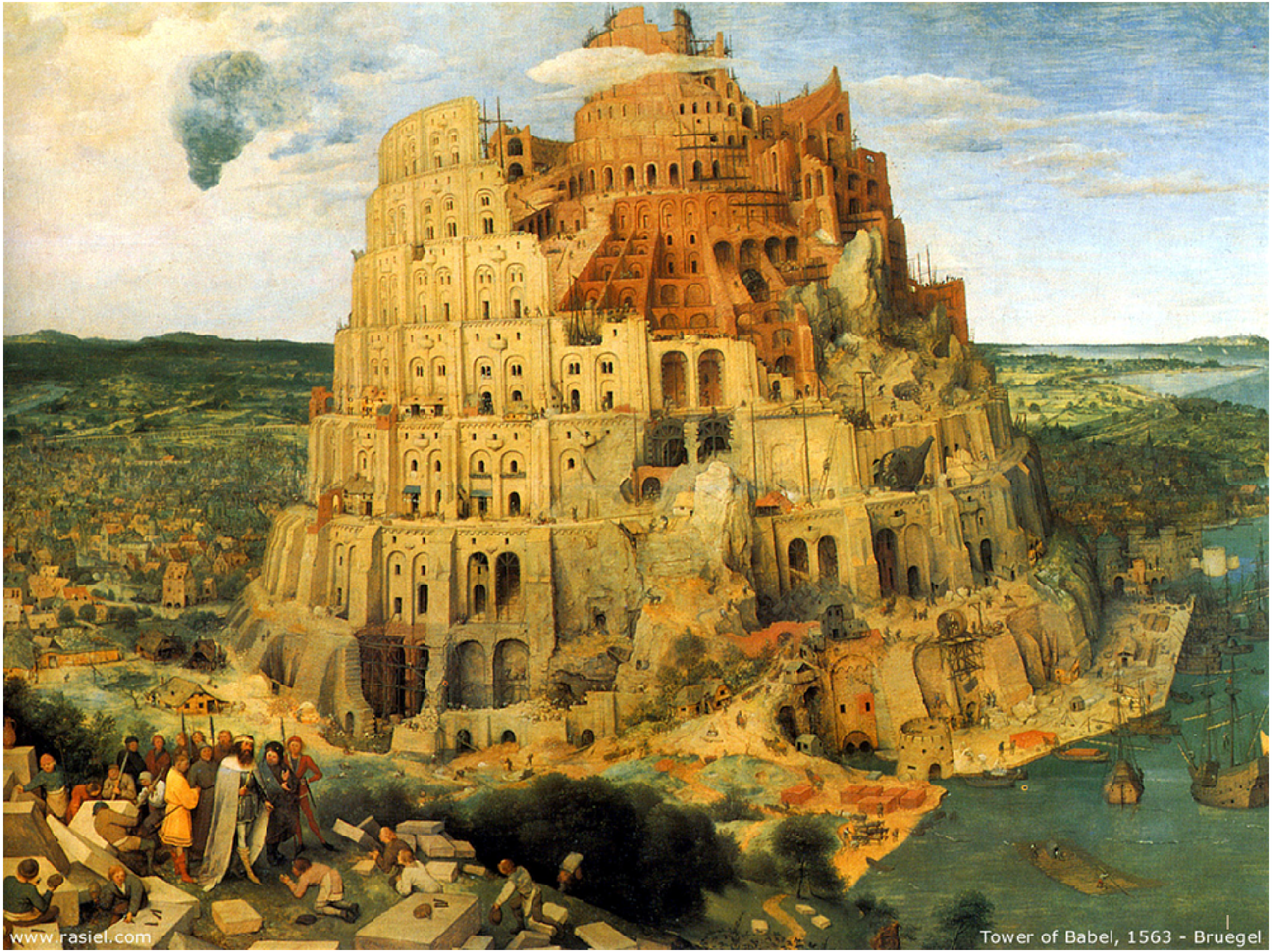
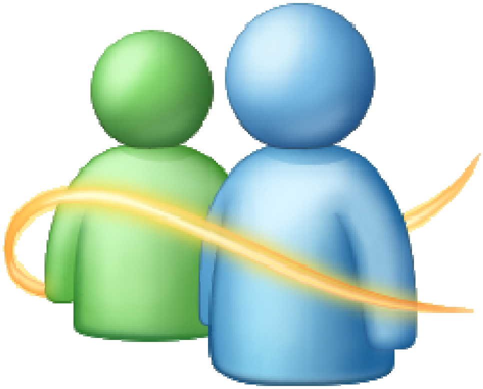
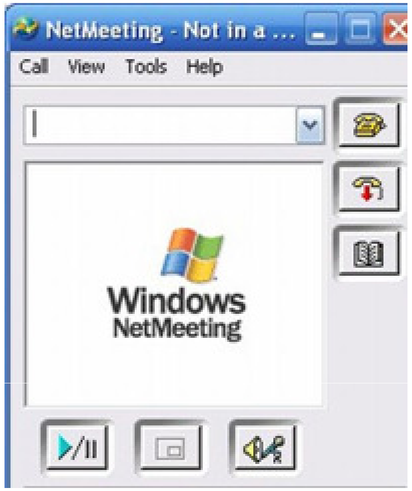


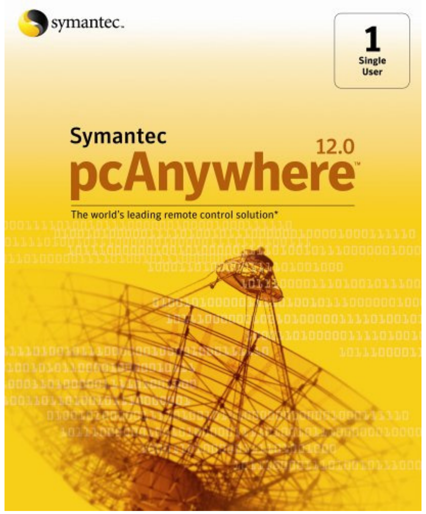
www.rasiel.com

Tower of Babel, 1563 - Bruegel

Collaboration support in SW







Google Docs

Google wave

eclipse
THE ECLIPSE PROJECT

Collaborative Real Time Coding

Stas Levin

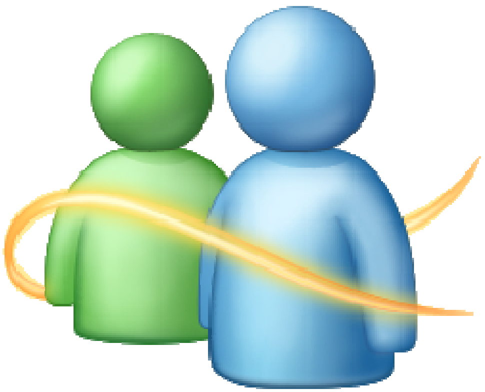
Advanced Software Tools Seminar, Tel Aviv University

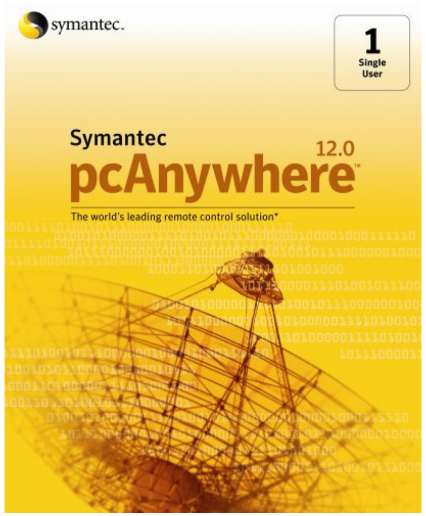
March 28, 2011

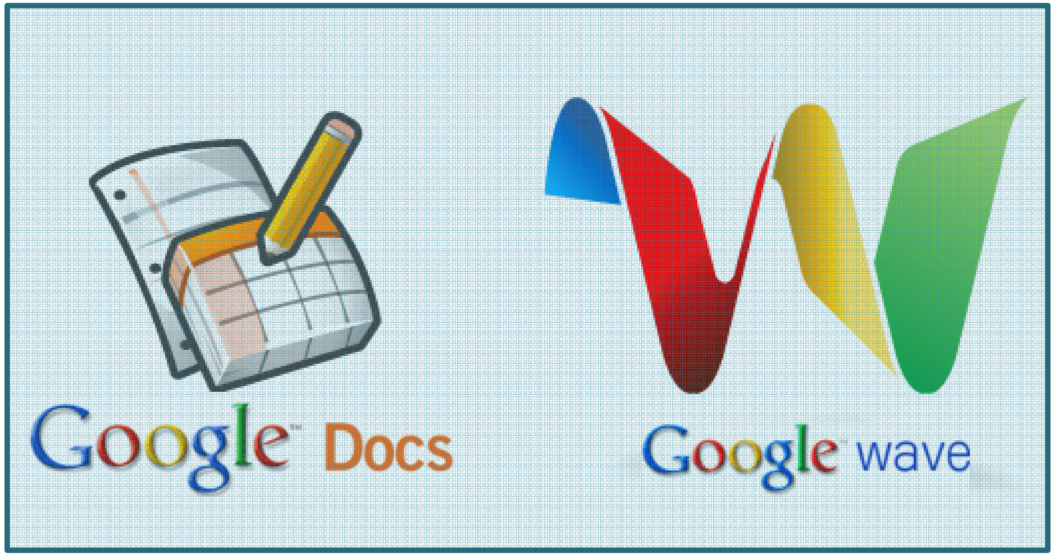
Agenda

- Collaborative software & SCM systems
- The CRTC approach
- A CRTC prototype
- Future directions
- Q&A

Collaboration support in SW





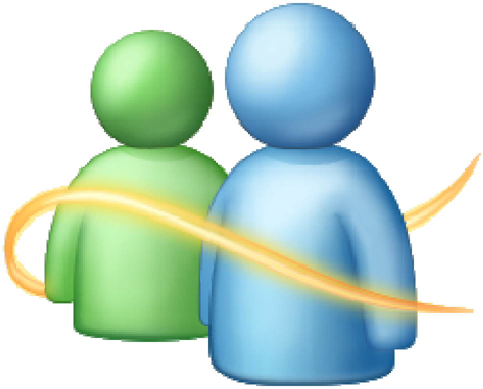


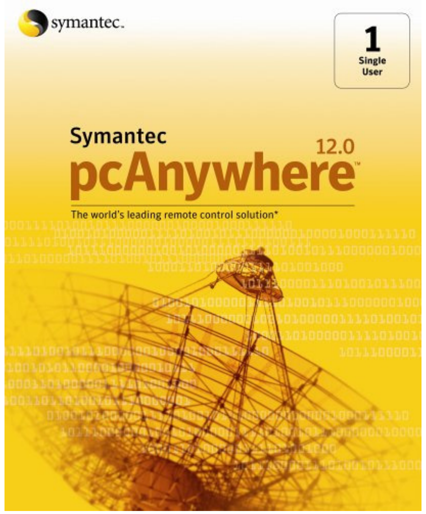
Collaborative Text Editing

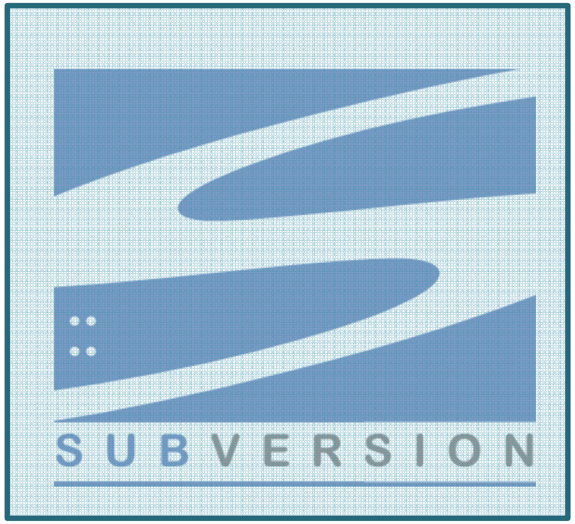
- Multiple users concurrently edit the same document, in real time
- All changes are visible to all users, in real time
- Group awareness mechanisms

Real Time Collaboration

Collaboration support in SW





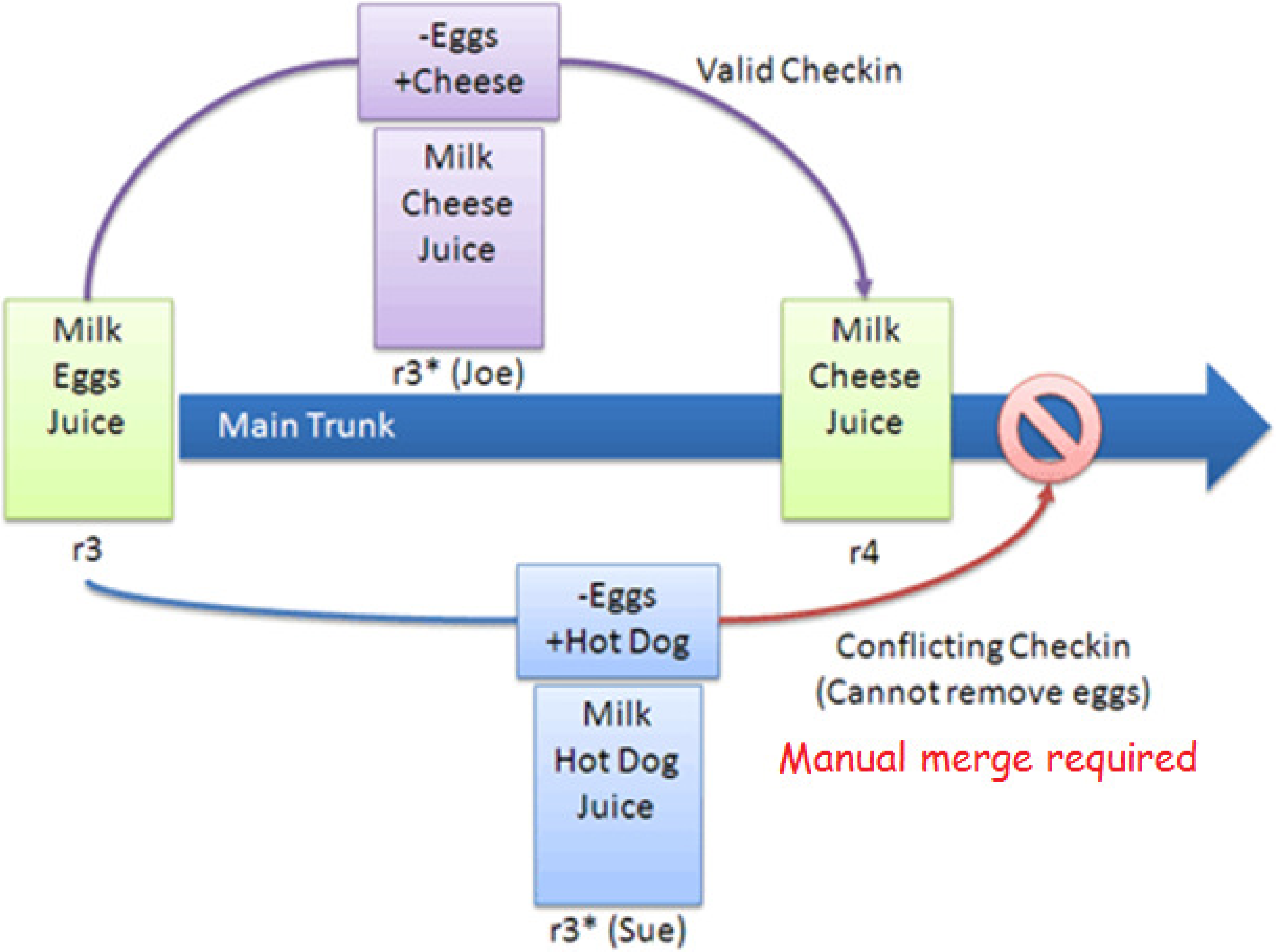


Software Control Management

"On any team project, a certain degree of confusion is inevitable. The goal is to minimize this confusion so that more work can get done … Configuration management is the art of … and controlling modifications to the software being built by a programming team. The goal is to maximize productivity by minimizing mistakes."

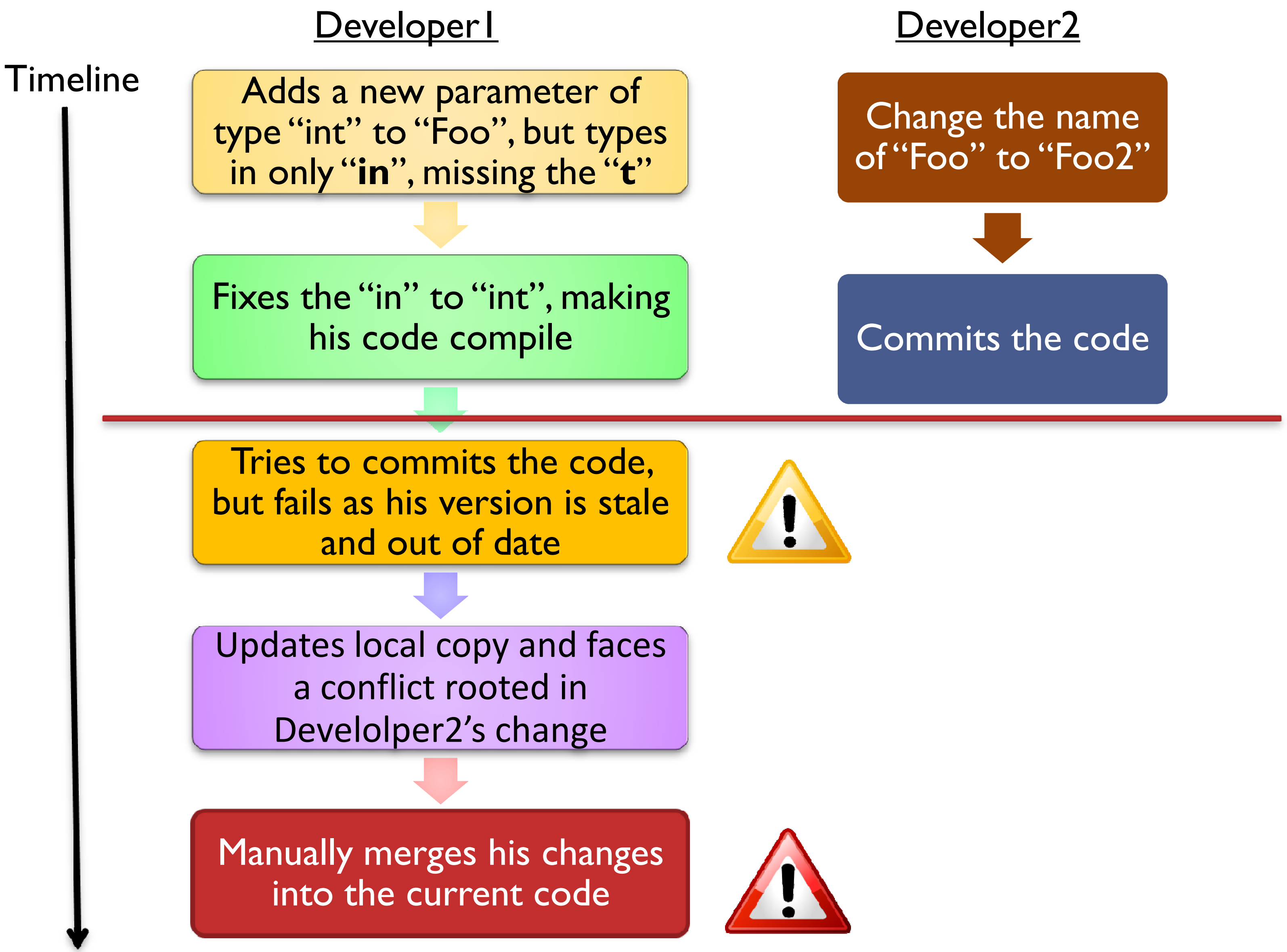
- Wayne Babich
- Software Configuration Management: Coordination for Team Productivity. Addison-Wesley, 1986.

The Dreaded Merge



http://betterexplained.com/articles/a-visual-guide-to-version-control/

A Use Case – current SCM



Developer1

Developer2

Timeline

Adds a new parameter of type "int" to "Foo", but types in only "**in**", missing the "**t**"

Fixes the "in" to "int", making his code compile

Change the name of "Foo" to "Foo2"

Commits the code

Tries to commits the code, but fails as his version is stale and out of date ⚠

Updates local copy and faces a conflict rooted in Develolper2's change

Manually merges his changes into the current code ⚠

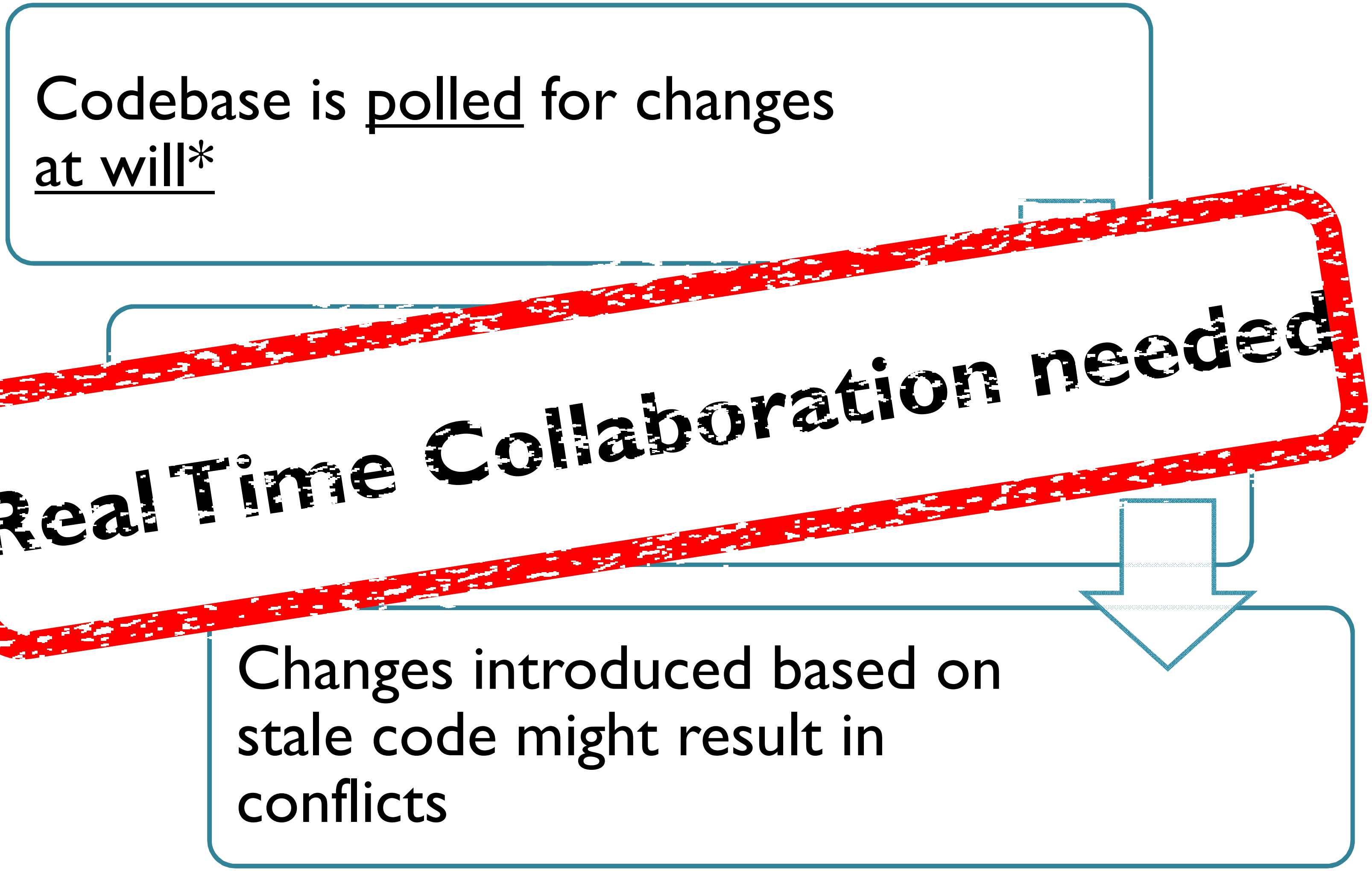
Manual Merge is Bad

- Time consuming, may require:
  - Understanding code written by others
  - Efforts from multiple developers
  - Regression tests

- Error prone, may:
  - Introduce (new?) misbehaviors
  - Leave an entire feature or parts of it out

For further information on the merge process:
"A State-of-the-Art Survey on Software Merging", Tom Mens

Conflict Promoters

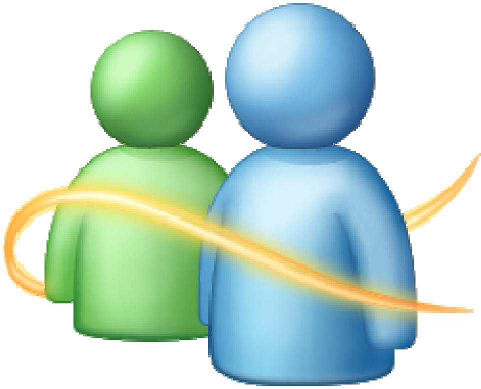


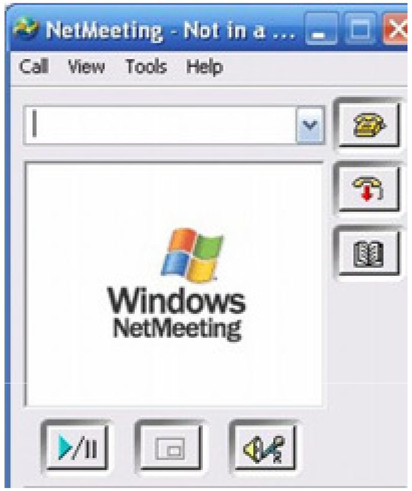
Codebase is polled for changes at will*

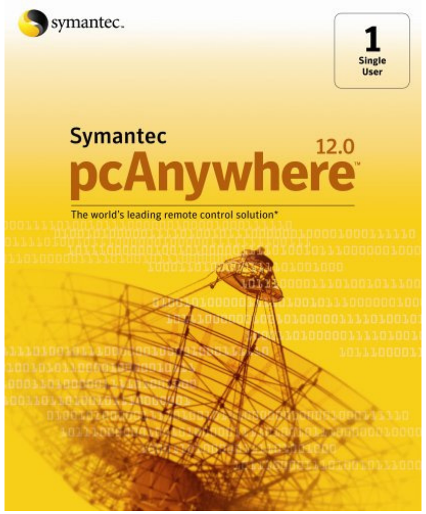
Real Time Collaboration needed

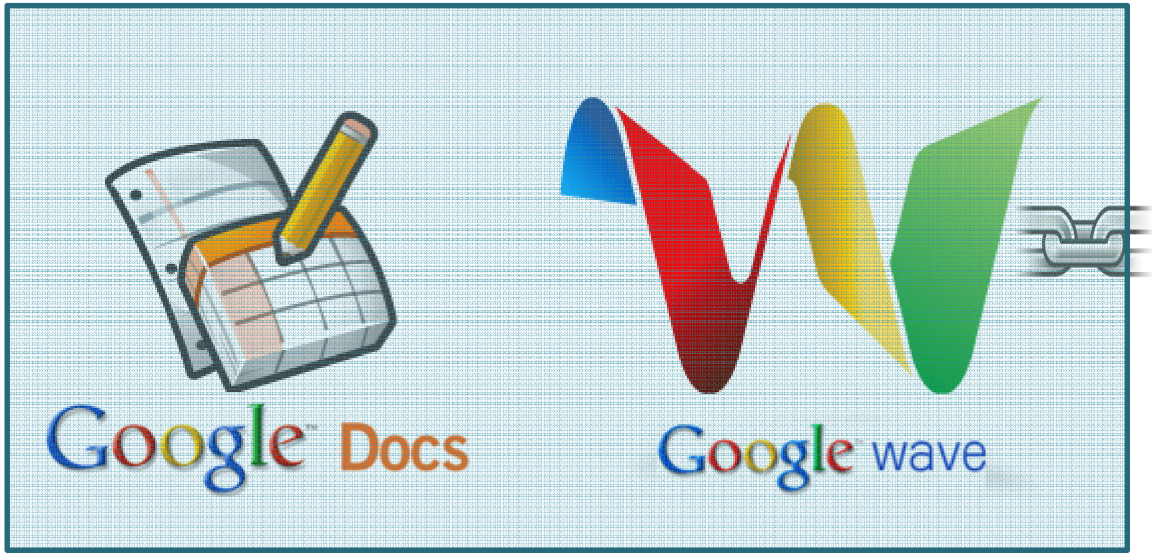
Changes introduced based on stale code might result in conflicts

Collaboration support in SW









SUBVERSION

Collaborative Real Time Coding

Stas Levin

Advanced Software Tools Seminar, Tel Aviv University

March 28, 2011

CRTC to the rescue

- Improve collaboration between developers working on same codebase

1. Eliminate merges incurred by conflicts
2. Keep developers' code up to date
3. Increase mutual awareness
4. …

Concurrent access control

- Optimistic locking
  1. Execute, don't lock
  2. Handle conflict

- Pessimistic locking
  1. Exclusively lock
  2. Execute operation

Responsiveness vs. Data integrity

The Challenges

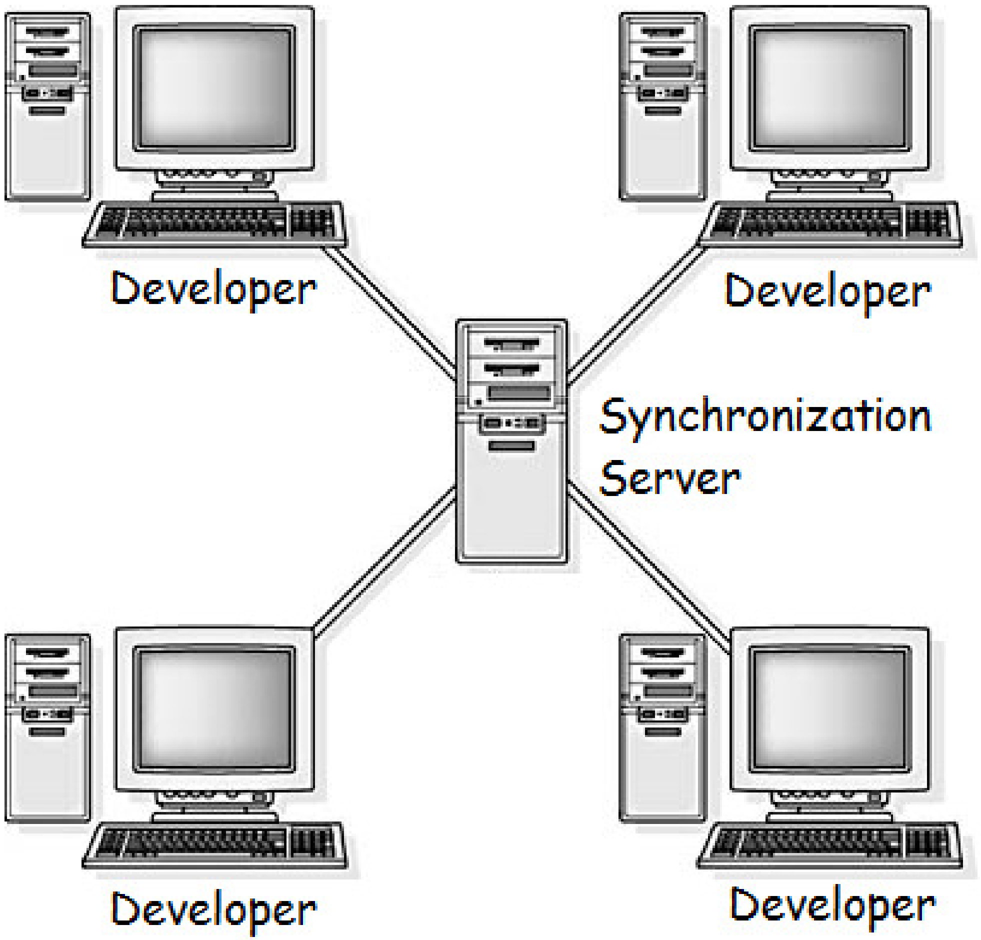
- How does one propagate code?
  - Files
  - Characters
  - Elements (methods, classes, etc.)

- When is the code propagated
  - Periodically
  - Upon certain events (save, build, etc.)

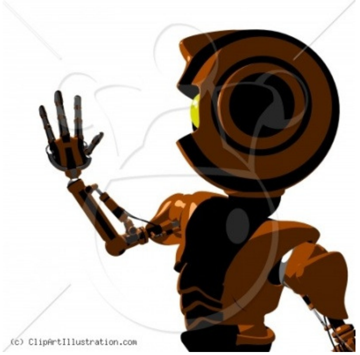
Suggested CRTC approach

- How does one propagate code?
  - Elements (methods, classes, etc.)
- When is the code propagated
  - Upon a successful background build
- Pessimistic, but granular locking scheme:
  - The locking scheme operates on semantic elements (e.g. methods, fields etc.)
  - Elements are pessimistically locked
  - Granularity promotes lower conflict rates

CRTC system architecture

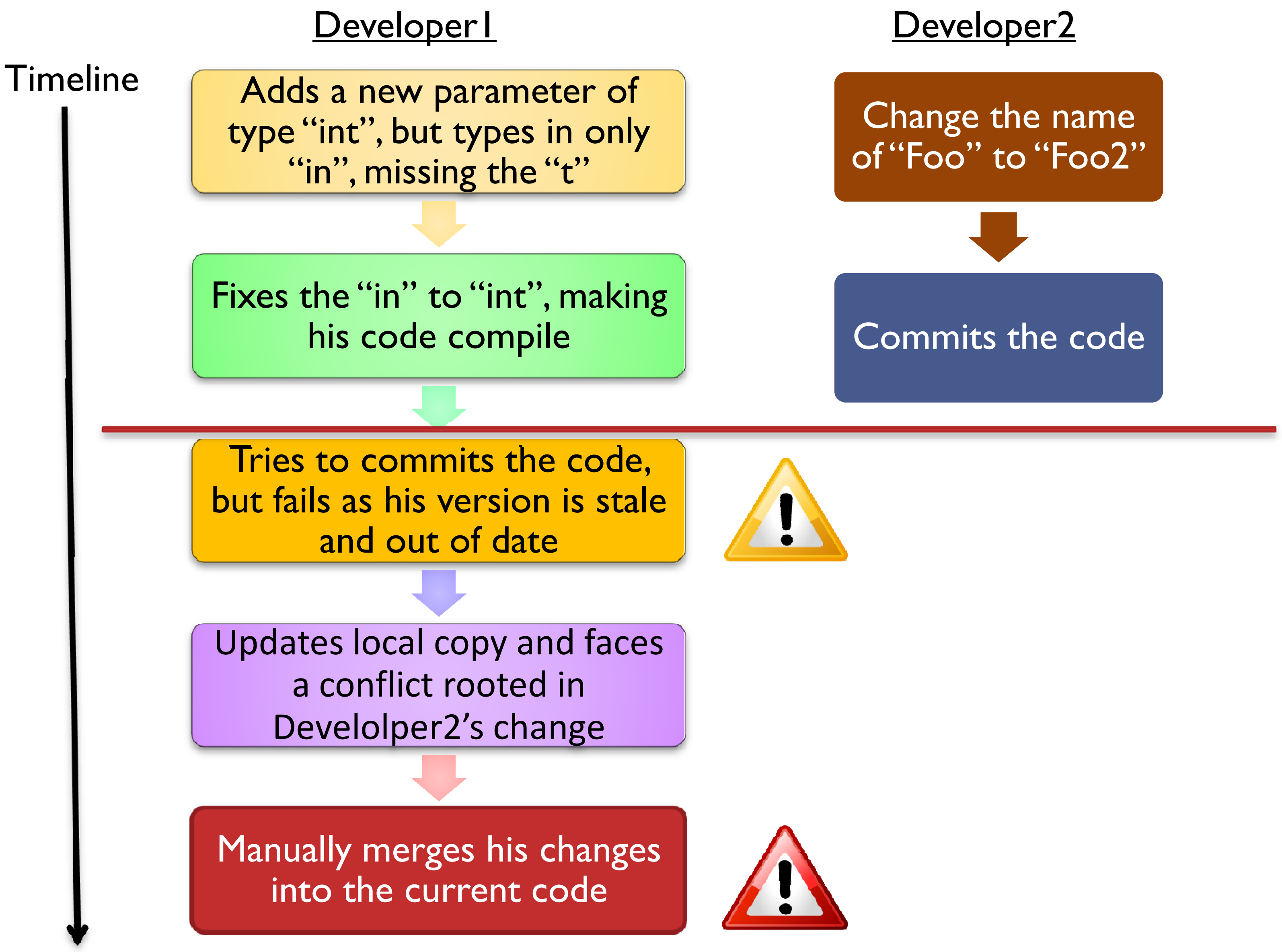


The Prototype



- Plug in for the Eclipse IDE
- Written in Java
- Employing Eclipse SDK
- Using Spring
- Tested in a VM environment

A Use Case – current SCM



Developer1

Developer2

Timeline

Adds a new parameter of type "int", but types in only "in", missing the "t"

Fixes the "in" to "int", making his code compile

Tries to commits the code, but fails as his version is stale and out of date

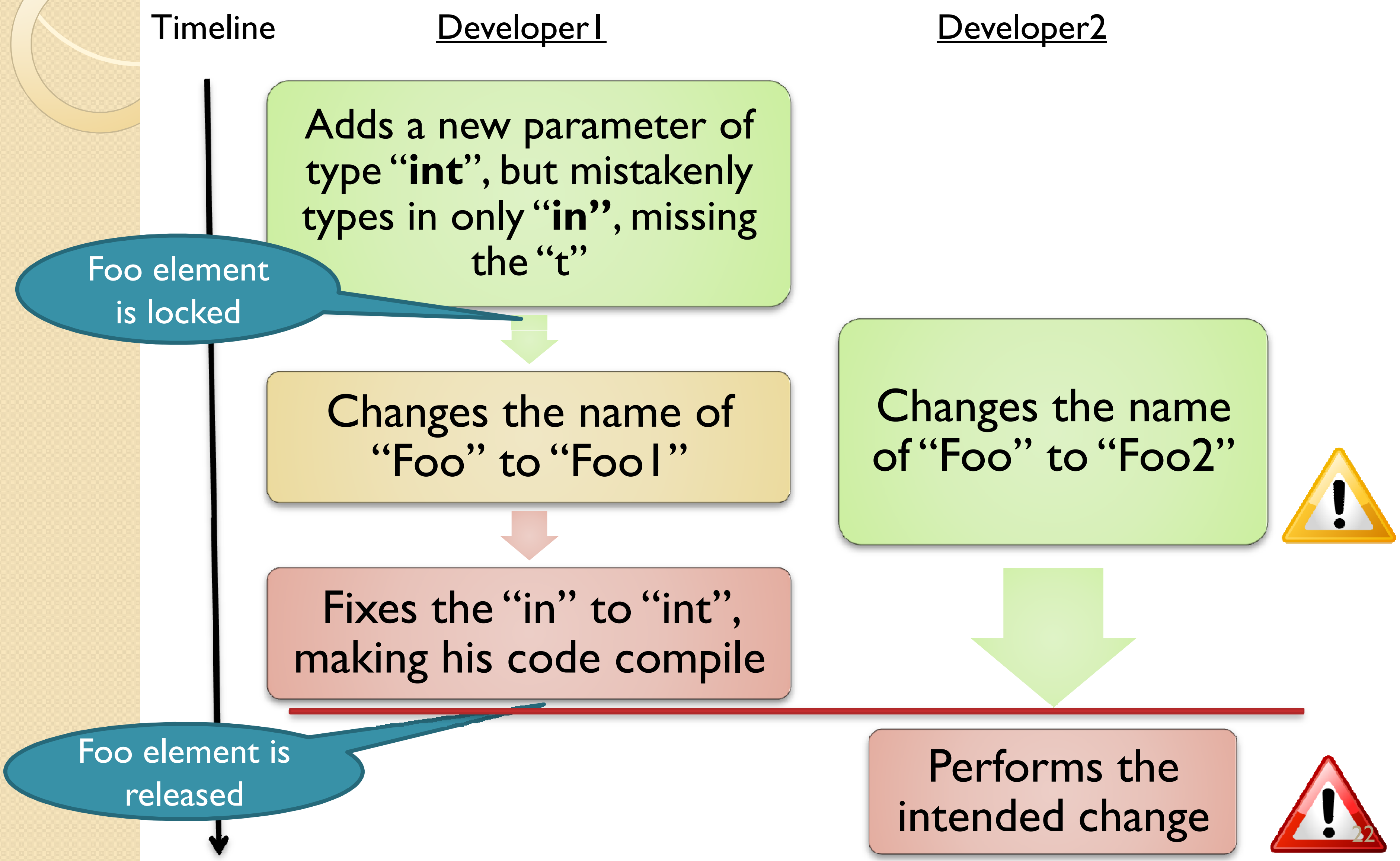
Updates local copy and faces a conflict rooted in Develolper2's change

Manually merges his changes into the current code

Change the name of "Foo" to "Foo2"

Commits the code

A Use Case – using CRTC



Timeline

Developer1

Developer2

Adds a new parameter of type "**int**", but mistakenly types in only "**in"**, missing the "t"

Foo element is locked

Changes the name of "Foo" to "Foo1"

Fixes the "in" to "int", making his code compile

Foo element is released

Changes the name of "Foo" to "Foo2"

Performs the intended change

CRTC in action

DEMO

Future Directions

- Redesigning SCM into CRTC

- CRTC & existing methodologies
  - Agile
  - Waterfall

- New methodologies
  - Development process
  - Coding conventions
  - Refactoring guidelines

QUESTIONS

Summary

- We've introduced a new concept called Collaborative Real Time Coding (CRTC)
- CRTC aims at:
  - reducing the need for manual merges
  - Generally boosting collaboration in software development teams
- We've demonstrated a prototype for a CRTC system

"It is the long history of humankind (and animal kind, too) those who learned to collaborate and improvise most effectively have prevailed." - Charles Darwin

stanisl@post.tau.ac.il