

# Collaboration, Leadership, Control and Conflict Negotiation in the NetBeans.org Community

Chris Jensen and Walt Scacchi

Institute for Software Research

School of Information and Computer Science

University of California, Irvine

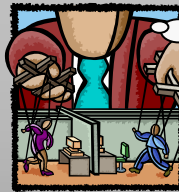
{cjensen, wscacchi}@ics.uci.edu

## Overview

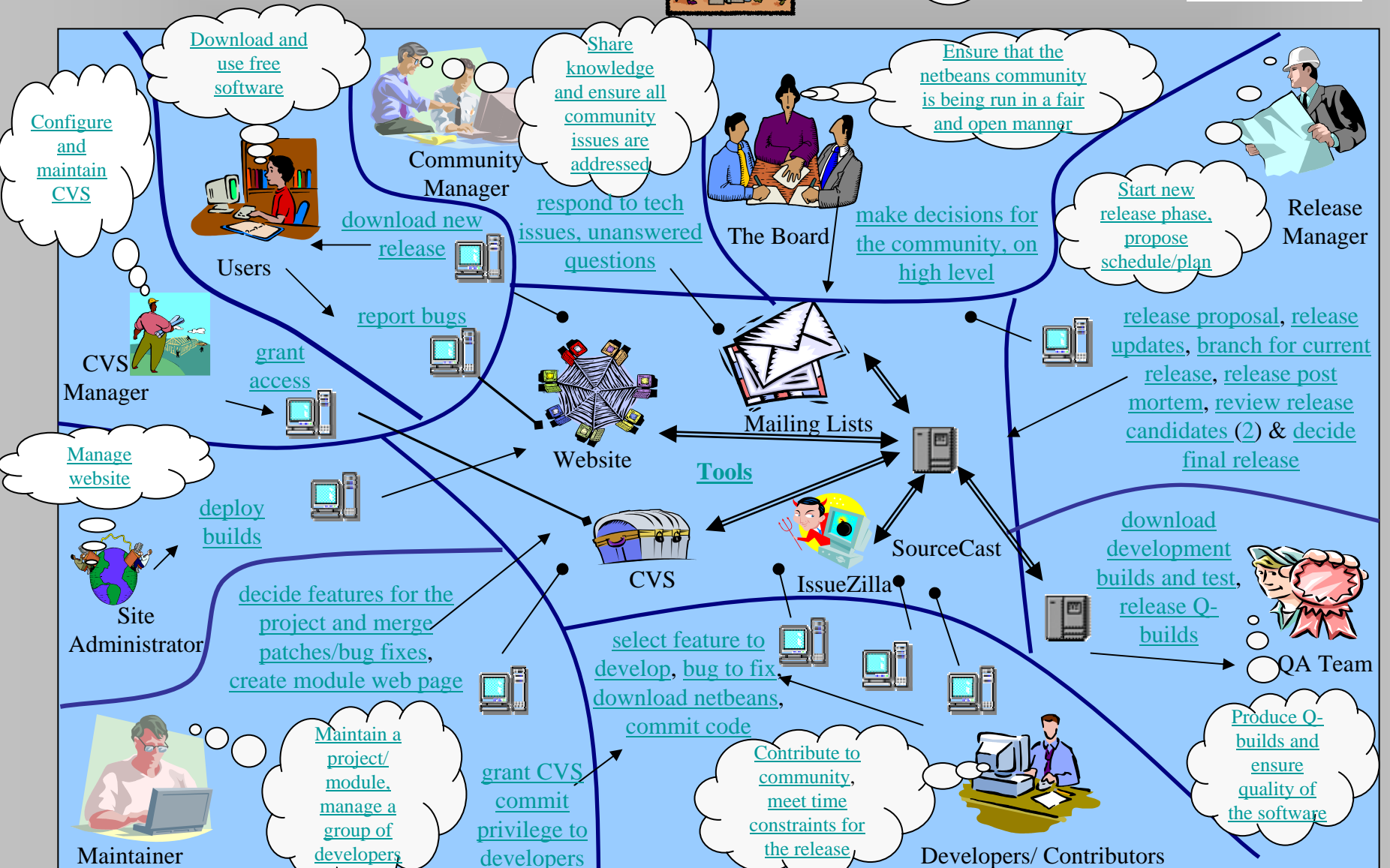
- OSSD Processes in NetBeans.org
- Examining OSSD processes within and across multiple OSS projects
- Emerging patterns of collaboration, leadership, control, and conflict negotiation
- Conclusions

## Understanding OSS Development Processes and Practices in NetBeans.org

- Discovering and modeling of open source software development processes
  - Phase1: reference model
  - Phase2: project history: development in context
  - Phase3: process-fragment mining
    - relevancy issues
  - Phase4: multi-technique process modeling, and simulated/remote process re-enactment
  - Phase5: analysis
- C. Jensen and W. Scacchi, [Discovering, Modeling, and Re-enacting Open Source Software Development Processes](#), Institute for Software Research, March 2004.



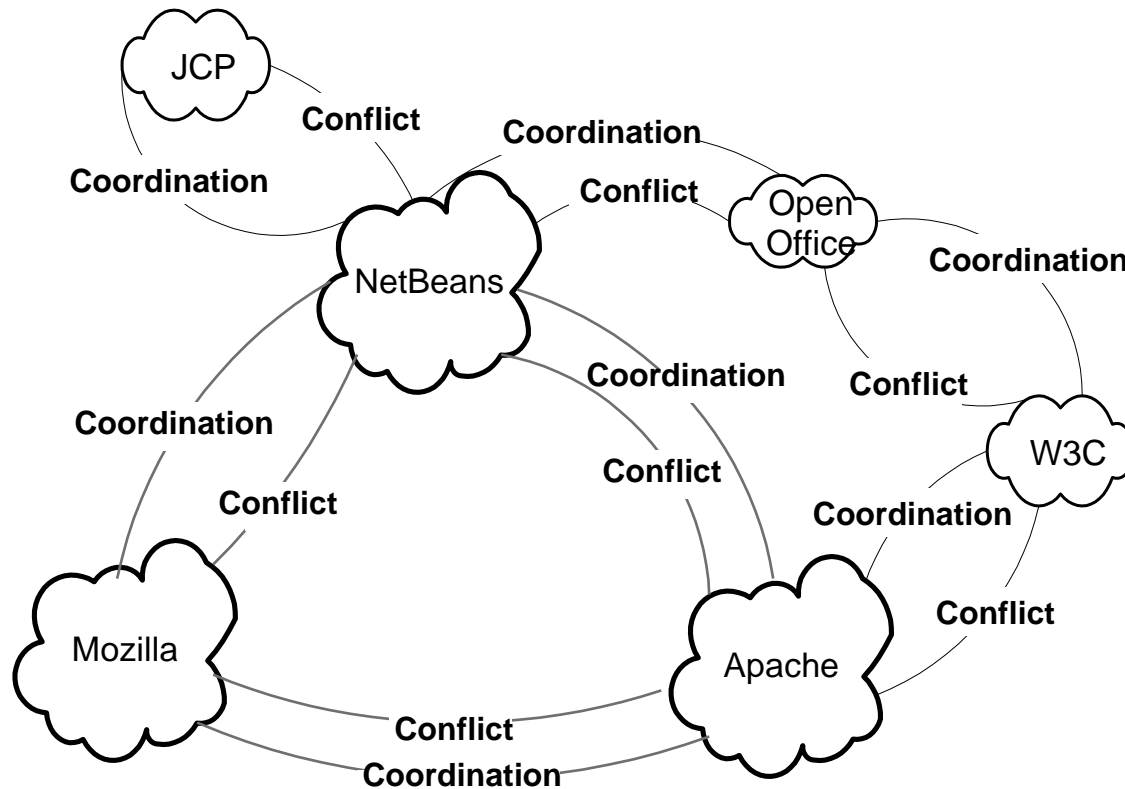
Funds, support,  
Promote  
Java/Open  
source



## OSSE 2004

- What we have since discovered and modeled by examining *multiple* OSSD processes in *interrelated* OSSD projects.
- NetBeans.org, Mozilla.org, Apache.org, BioBeans.org, Tigris.org, Java Tool Community, etc.
- Leadership and control sharing within and across individuals and organizations are common source of conflict in OSSD projects.

# NetBeans.org Software Project Ecosystem



## Objects of Interaction

- Development artifacts (“software informalisms”)
- Protocols
  - HTTP, RPCs
- Shared data formats
  - HTML, XML, CGI
- Community infrastructure tools
  - Defect repositories (e.g. Bugzilla), Collaborative development tools (e.g. WIKI, CVS, mail list managers)
- Product infrastructure
  - Plug-ins, Modules
- OSS development processes

## Intra-community issues

- Collaboration
  - Guidelines and policies
    - Development tasks; style guidelines; public floggings
  - Separation of concerns:
    - architectural strategy (plug-ins) for collaborative success;
    - freedom of extension/expression through contributed source code--reduces involvement with socio-political project issues
    - Volunteer versus salaried developers--collaboration breakdowns lead to product failures



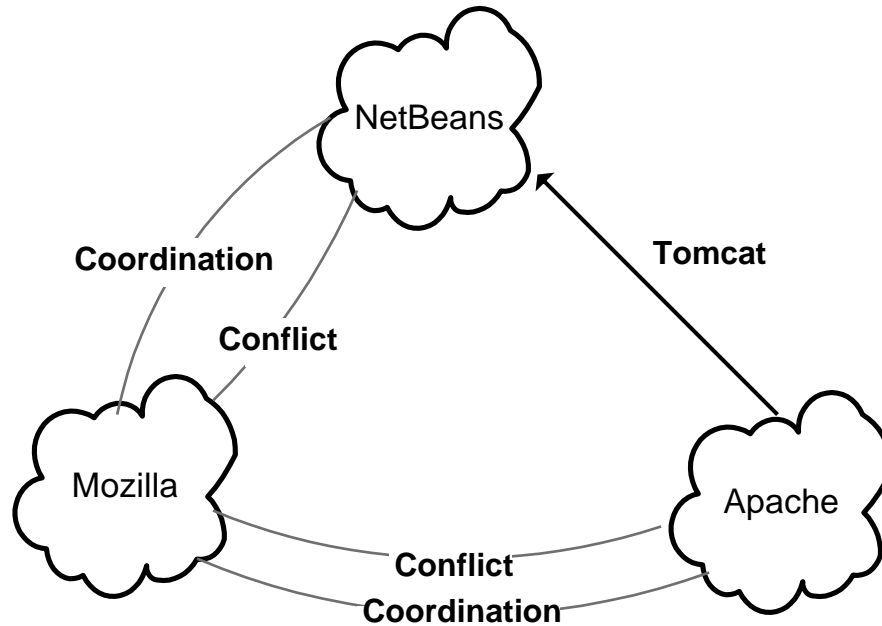
## Intra-community issues

- Leadership and Control
  - Accountability and expectations based on precedent and volunteerism
  - Transparency in decision-making
    - Project “management” limited to coordinating roles
  - Consent in decision-making
    - Many contributors assume consensus decision-making, and breakdowns arise when Sun asserts prerogative
- Conflict Resolution
  - Not face-to-face
  - Generally done in “public” via discourse transactions on discussion lists, else turned over to community governance board for resolution.

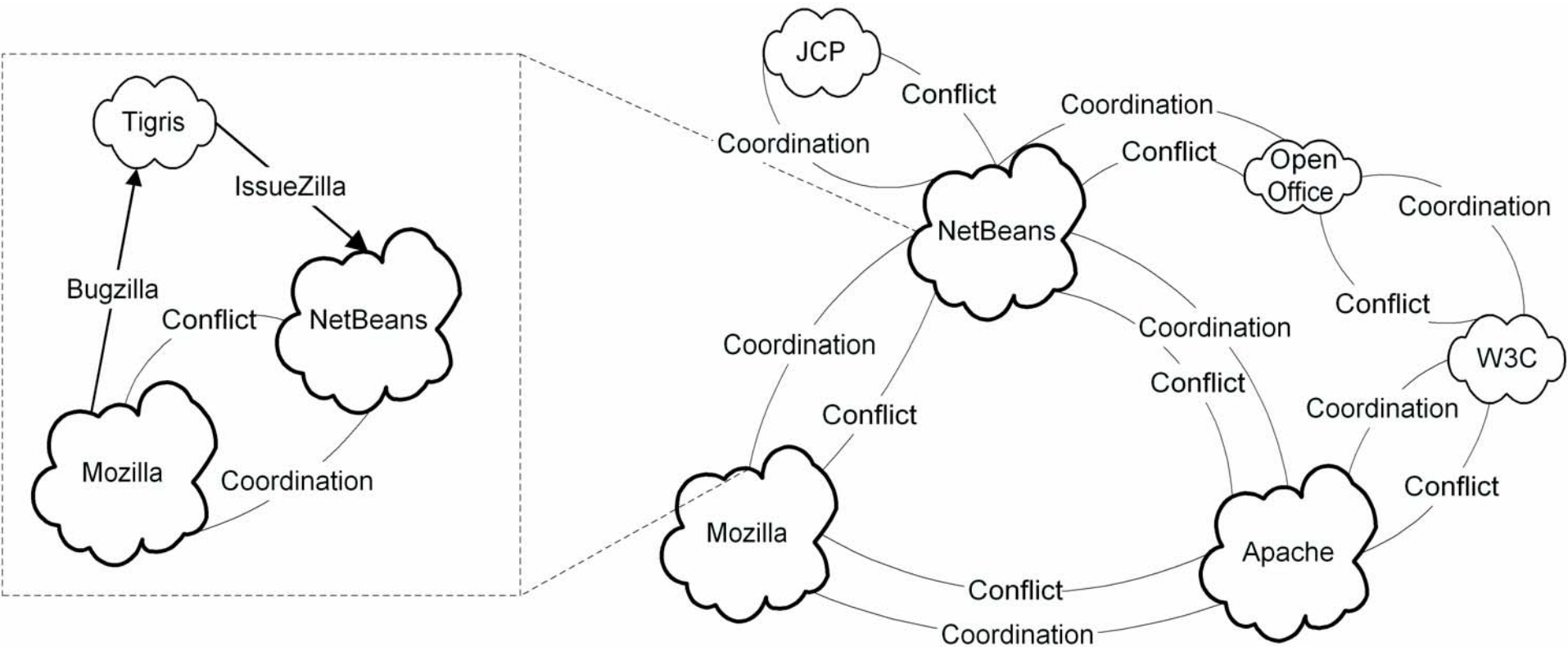
## Inter-community issues

- Communication and collaboration
  - Bug reports and feature requests
  - Patches submitted
  - Java.net, Java Tools Community, and Java Community Process;
- Leadership and control
  - Sun NetBeans + IBM Eclipse ???
- Conflict resolution
  - Mailing lists; Slashdot; Developer blogs

## Direct Interaction



# Indirect process interactions across projects



## Interaction Patterns

- Patterns can be detected and include:
  - Integration of a tool or support for a technology created by another community
  - Defect detection and reduction
    - Organizations contribute defect reports/patches detected in another organization's tool or technology implementation
  - Infrastructure evolution planning
    - Research contributing to discussions of future/changes in tools and technologies
      - Discovery, assessment of effects on one's own community
- These interactions give rise to additional opportunities for coordination and conflict

## Conclusions

- OSSD processes occur within and across multiple projects spanning the Internet infrastructure
- Multiple project/organizational interaction may be coordinative or conflictive
- Interaction is driven by ongoing synchronization and stabilization of objects of interaction across the Internet infrastructure
- Project interaction patterns are emerging, detectable, modeled, and suitable for simulated re-enactment
- Modeling processes within and across multiple dependent projects is new, challenging, and important.

## References

see <http://www.isr.uci.edu/research-open-source.html>

- C. Jensen and W. Scacchi, Discovering, Modeling, and Reenacting Open Source Software Development Processes, Institute for Software Research, March 2004.
- C. Jensen and W. Scacchi, Process Modeling the Web Information Infrastructure, *Proc. 5th. Software Process Simulation and Modeling Workshop*, Edinburgh, Scotland, May 2004.
- W. Scacchi, [Understanding the Requirements for Developing Open Source Software](#), *IEE Proceedings--Software*, 149(1), 24-39, 2002.
- W. Scacchi, [When is Free/Open Source Software Development Faster, Better, and Cheaper than Software Engineering?](#) Working Paper, Institute for Software Research, UC Irvine, April 2003.
- W. Scacchi, [Free/Open Source Software Development Practices in the Computer Game Community](#), *IEEE Software*, Special Issue on Open Source Software, 21(1), 59-67, January-February 2004.
  
- This presentation will be found at:  
<http://www.ics.uci.edu/~wscacchi/Presentations/ProSim04/>

ISR Research - Open Source Software Development - Mozilla

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop <http://www.isr.uci.edu/research-open-source.html> Search Print

## ISR Institute for Software Research

UNIVERSITY OF CALIFORNIA, IRVINE

[Home](#) | [People](#) | [Research](#) | [Publications](#) | [Tech Transition](#) | [Events](#) | [Partnerships](#) | [About ISR](#)

### Open Source Software Development

**Overview** UCI research in open source software development focuses on empirically-based studies of the processes, practices, and communities that

**Analysis and Testing** develop open source software. Ethnographic and virtual ethnographic research methods are employed in the field studies of open source software development

**Architecture** in communities that include those centered on Internet infrastructure, X-Ray astronomy and deep space imaging, networked computer games, and academic software design research.

**Acquisition and Ecommerce**

**Computer-Supported Cooperative Work**

**Configuration Management**

**Education**

**Environments**

**Human-Computer Interaction**

**Hypermedia**

**Information Visualization**

**Internet-scale Event Notification**

**Open Source Software**

#### Faculty

Walt Scacchi

#### Collaborators

Mark Ackerman, University of Michigan, Ann Arbor  
Margaret Elliott, UCI ISR  
Justin Erenkrantz, UCI ISR  
Les Gasser, University of Illinois, Urbana-Champaign  
Chris Jensen, UCI ISR  
John Noll, Santa Clara University  
Jason Robbins, UCI ISR  
Richard N. Taylor, UCI ISR  
Julia Watson, The Ohio State University

#### Projects

NSF ITR: Understanding Open Software Communities, Processes and Practices: A Socio-Technical Approach ([award abstract](#))  
NSF ITR: Collaborative Research: Organizational Dynamics of Software Problems, Bugs, Failures, and Repairs ([award abstract](#))  
NSF IIS: Collaborative Research: Research Directions for Continuous (Re)Design in Free/Open Source Software Systems ([award abstract](#))



# Acknowledgements

- *Project collaborators:*
  - Mark Ackerman, UMichigan, Ann Arbor
  - Les Gasser, UIllinois, Urbana-Champaign
  - John Noll, Santa Clara University
  - Margaret Elliot, Chris Jensen, UCI-ISR
  - Julia Watson, The Ohio State University
- *Funding support:*
  - National Science Foundation, ITR#0083075, ITR#0205679, ITR#0205724, and ITR#0350754.
  - No endorsement implied.

## References

see <http://www.isr.uci.edu/research-open-source.html>

- Elliott, M. and Scacchi, W., Free Software Development: Cooperation and Conflict in A Virtual Organizational Culture, in S. Koch (ed.), *Free/Open Source Software Development*, Idea Publishing, to appear, 2004.
- C. Jensen and W. Scacchi, Discovering, Modeling, and Reenacting Open Source Software Development Processes, Institute for Software Research, March 2004.
- C. Jensen and W. Scacchi, Process Modeling the Web Information Infrastructure, *Proc. 5th. Software Process Simulation and Modeling Workshop*, Edinburgh, Scotland, May 2004.
- W. Scacchi, [Understanding the Requirements for Developing Open Source Software](#), *IEE Proceedings--Software*, 149(1), 24-39, 2002.
- W. Scacchi, [Free/Open Source Software Development Practices in the Computer Game Community](#), *IEEE Software*, Special Issue on Open Source Software, 21(1), 59-67, January-February 2004
- This presentation will be found at:  
<http://www.ics.uci.edu/~wscacchi/Presentations/OSSE04/>

## References

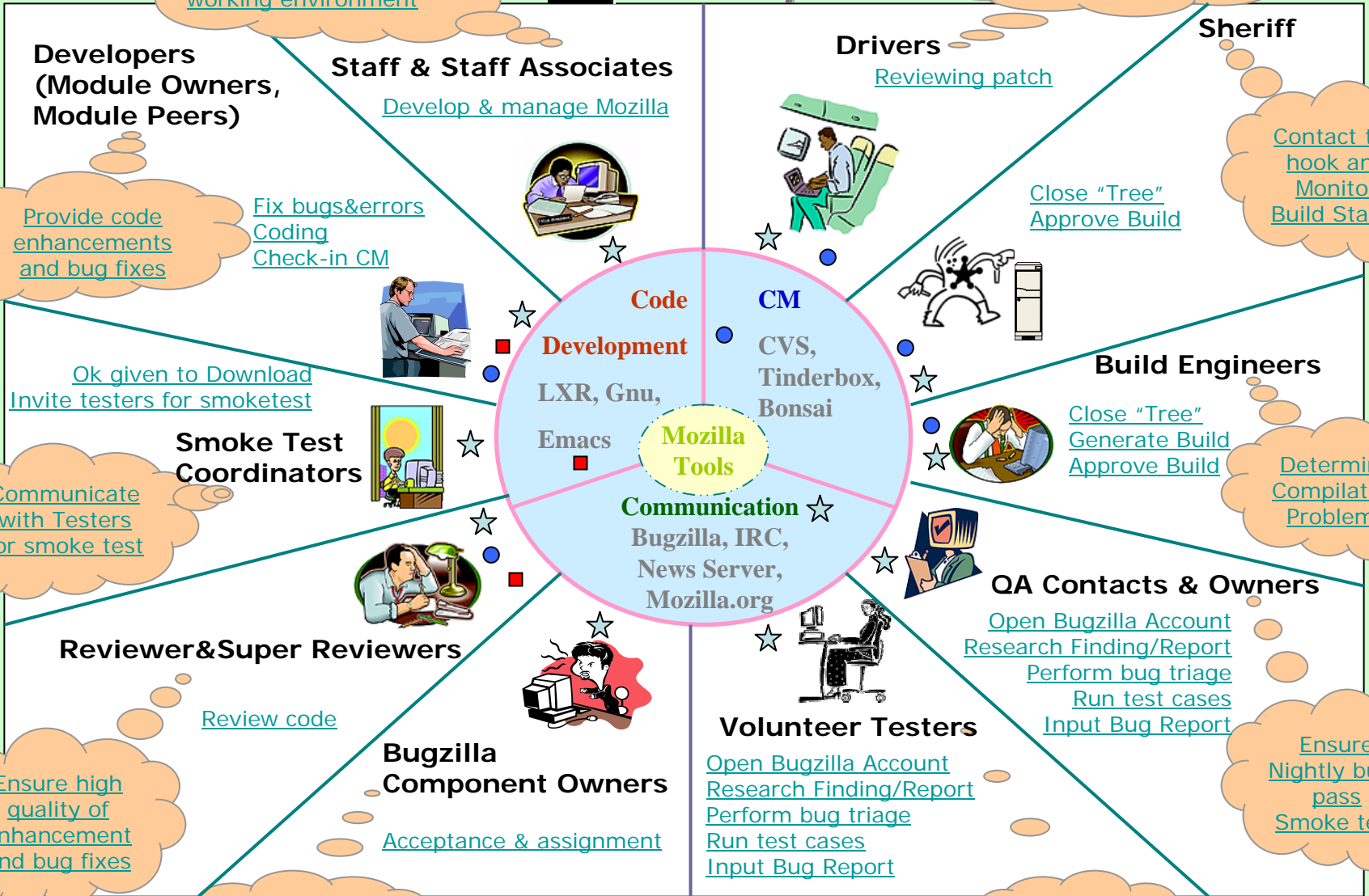
- 
- A. Hars and S. Ou, Working for free? Motivations for participating in open source projects, *International Journal of Electronic Commerce*, 6(3), Spring 2002.
- G. Madey, V. Freeh, and R. Tynan, Modeling the F/OSS Community: A Quantitative Investigation, in *Free/Open Source Software Development*, in Stephan Koch, (ed.) Idea Publishing, forthcoming.
- D.M. Nichols & M.B. Twidale, The Usability of Open Source Software, *First Monday*, 8(1), January 2003.

# N Netscape

Provide Resource and Manpower

Collaborate and provide Mozilla working environment

Help to prioritize checkin patches



Contact the hook and Monitor Build Status

Determine Compilation Problems

Ensure Nightly build pass Smoke test

Help to reveal bugs

Help to assign bugs

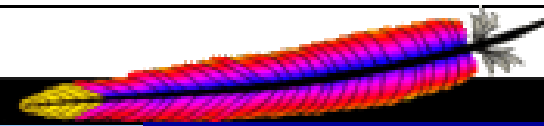
Ensure high quality of enhancement and bug fixes

Communicate with Testers for smoke test

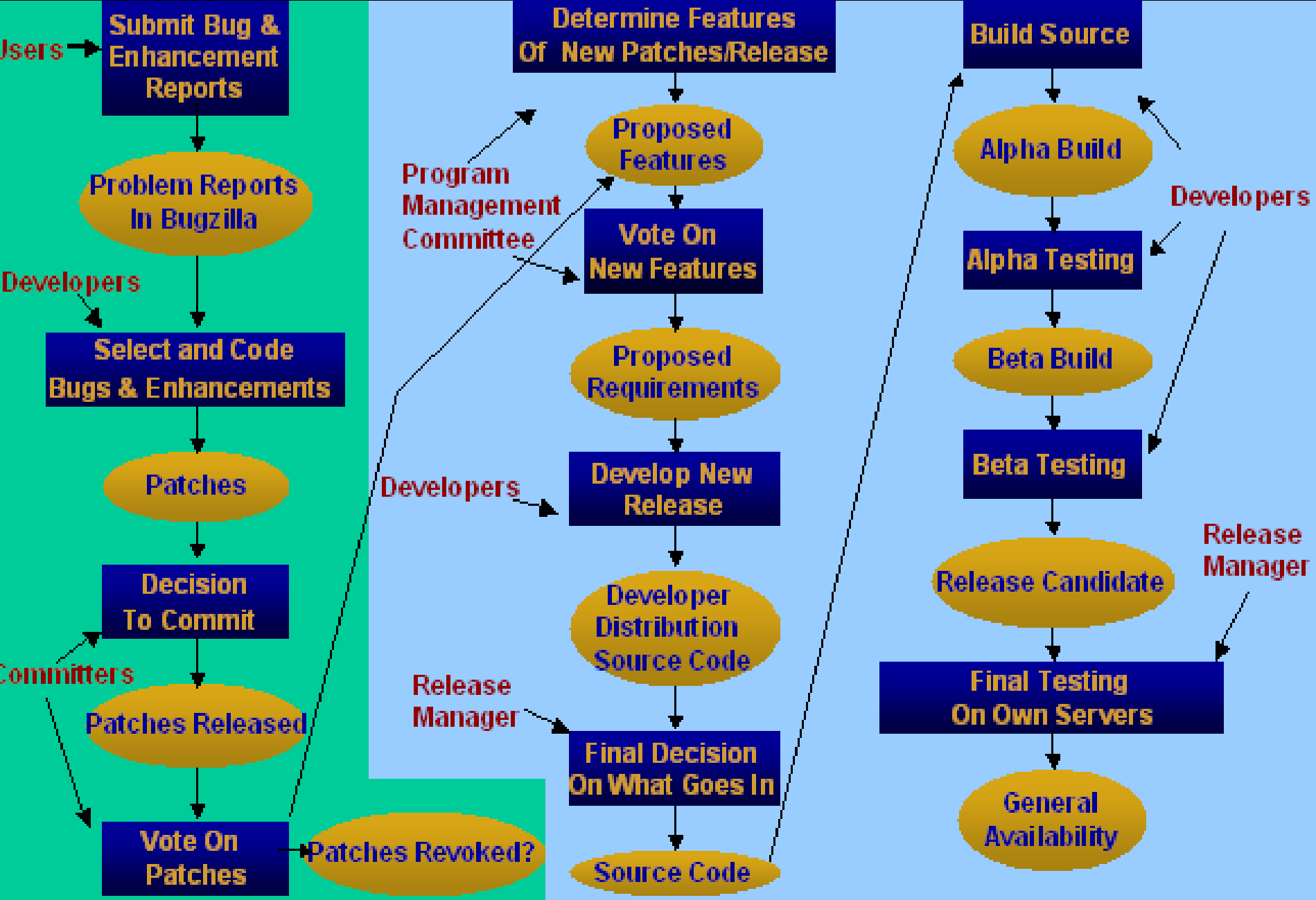
Ok given to Download Invite testers for smoketest

Provide code enhancements and bug fixes

# Apache HTTPD Release Process



# Apache HTTP SERVER PROJECT





Config +C +I - S Op C

frame	sub	sup	slx	isx	slt
Netbeans ...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Classes V

- :THING A
- :SYSTEM-CLASS A
  - Process Model (1)
  - Agent (9)
  - Resource (17)
  - Tool (10)
  - Action (23)
- Control Flow (1)
- Script

