git.io/auto
enterprise and open source projects face common challenges
collaboration among/between distributed teams
collaboration among/between distributed teams

*push for more rapid delivery*
collaboration among/between distributed teams

push for more rapid delivery

stewardship / ownership / maintenance
open source communities have largely solved for these problems
asynchronous communication

iterative development

ci & other automation

community & maintainers

reuse & package management
most organizations struggle with **awareness of existing enterprise development**
InnerSource applies lessons of open source inside enterprise environments
tenets of InnerSource
open
open

dararent
open
transparent
participatory
open
transparent
participatory
collaborative
open
transparent
participatory
collaborative

governed
Open repositories

• prior art
• reuse
• contribute
Open

repositories

collections

• create branches & propose changes
• governance
• branches and protections
source is an artifact of a production process
Software Development

- Operations
- Executive Leadership
- Security
- Domain Experts
- Design
- Project Management
- Customer Support
Transparent
Transparent

process

• documented processes
• communication preserved and open
“we should do X”

“we tried that a long time ago and it didn’t work”

“oh, ok”
“we should do X”

“we tried that a long time ago, see https://github.com/org/repo/pulls/123”

<<reading>>

“interesting, it looks like”
Transparent

process development

• collaborate while code is written
• surface concerns early
• reduce churn
a massive stack of information does not magically enable people to participate
users

contributors

maintainers
Participatory
Participatory

users

• discovery

• installation
Participatory

users

contributors
contributions

open an issue
improve docs
offer insight/feedback
unsolicited improvements
backlog help
Participatory

users

contributors

maintainers
maintainers

set vision and direction

document processes

set & maintain tone

approve / collaborate / deny submissions

facilitate shared ownership
an organization that designs a system will inevitably produce a design whose structure is a copy of the organization’s communication structure

- Melvin Conway
Collaborative
Collaborative informal teams
org/devGroupA  org/devGroupB  org/devGroupC
Collaborative informal teams

- seek help
- observe
- offer assistance
Governed
Governed

policies and procedures

• reporting issues
• contributing code
• code of conduct
Governed

policies and procedures

enforcement

• maintainers

• automated
a massive stack of information does not magically enable people to participate
help people find and use your project
adopt a single platform for code

• find things
• discussion & community
• reuse & contribution
• includes more than coders
adopt a single platform for code

provide tailored documentation

• end users
• contributors
• interested “public”
help people become contributors
reduce friction

• script all the things
• help new participants find ways to contribute
• create a welcoming community
reduce friction

provide constant feedback loops

• ci / cd
• quality standards
• coverage
• security
equip maintainers for success
hire a community leader

• support maintainers
• promote internal projects
• work with legal
• remove blockers
• engage open source communities
hire a community leader

automate

• issue, PR, and review assignment
• review
Thank you!

Visit GENIVI at http://www.genivi.org or http://projects.genivi.org
Contact us: help@genivi.org

GENIVI is a registered trademark of the GENIVI Alliance in the USA and other countries.
Copyright © GENIVI Alliance 2017.