eclipseCON™ 2008

# Using ECF for Lightweight Distributed Team Collaboration

Scott Lewis - ECF Project Lead
Markus Kuppe – ECF Committer

http://www.eclipse.org/ecf

# Developer Team Collaboration

# Teams they are a changin'

- Membership
  - Distributed, Multiple Groups, Cross-organizational
  - Diverse Skills, Backgrounds, Cultures
  - Peers control membership
- Project Organization
  - Flat (ter), Voluntary, More team-driven
  - Team Members Have More Control
- Community Expectations
  - Open Planning
  - Responsiveness
    - Bug reports
    - Contributions
    - Newsgroups, Mailing Lists, IRC, etc.

# Implications for Team Communications Technology

- Interoperability
  - Multiple Communication Tools
  - People in multiple groups
  - Very dynamic team formation
- Integration
  - Fewer UIs -> Less Complexity -> More usability
  - Lightweight
- Extensibility
  - Team-specific Tools
  - Support for specific processes/approaches/conventions

# ECF Goal

**Lower Barriers to Team and Community Communication**

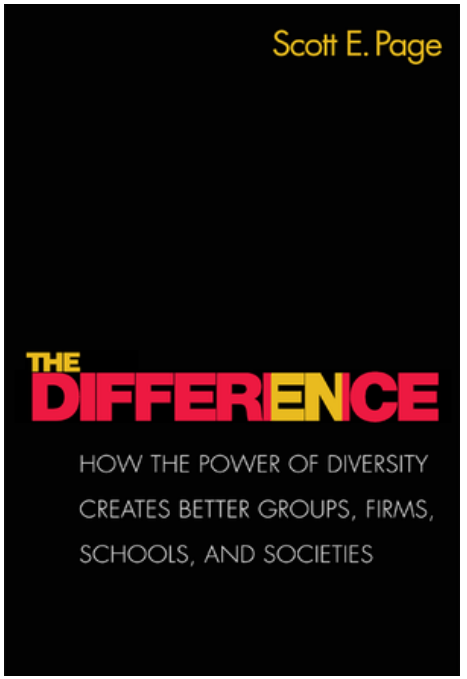
**by providing**

## Interoperable, Integrated, Extensible Framework

# Team Productivity: Diversity Trumps Ability

***The Difference: How the Power of Diversity Creates Better Groups, Firms, Schools, and Societies***

- Conditions
  - High-performing individuals
  - Difficult problems
- Good Diversity
- Bad Diversity



# Exploit Diversity

- Make it Easy for 'Outsiders' to Communicate/Contribute
  - Client Interoperability
- Communicate Publicly with Community
  - IRC, IM/Conference Calls, ECF Collaboration Groups, etc.
  - Solicit Contributions
    - Bugzilla/BugDay/IRC, etc.
- Expose Unfinished Work/Problems
  - Milestones, Dev Mailing List, Wiki, Bugzilla ALL GOOD
  - IM/Chat
  - Enable Opportunities for External Contributions

# Tools and UI Integration

- Existing Tooling
  - Workbench (sharing resources)
  - Bug Tracking (Mylyn sharing tasks)
  - Conferencing
  - Shared Editing
  - Build/Deploy Infrastructure (notifications, etc)
- New Eclipse-Based Tools/Products

# Demos – Interoperability and Integration

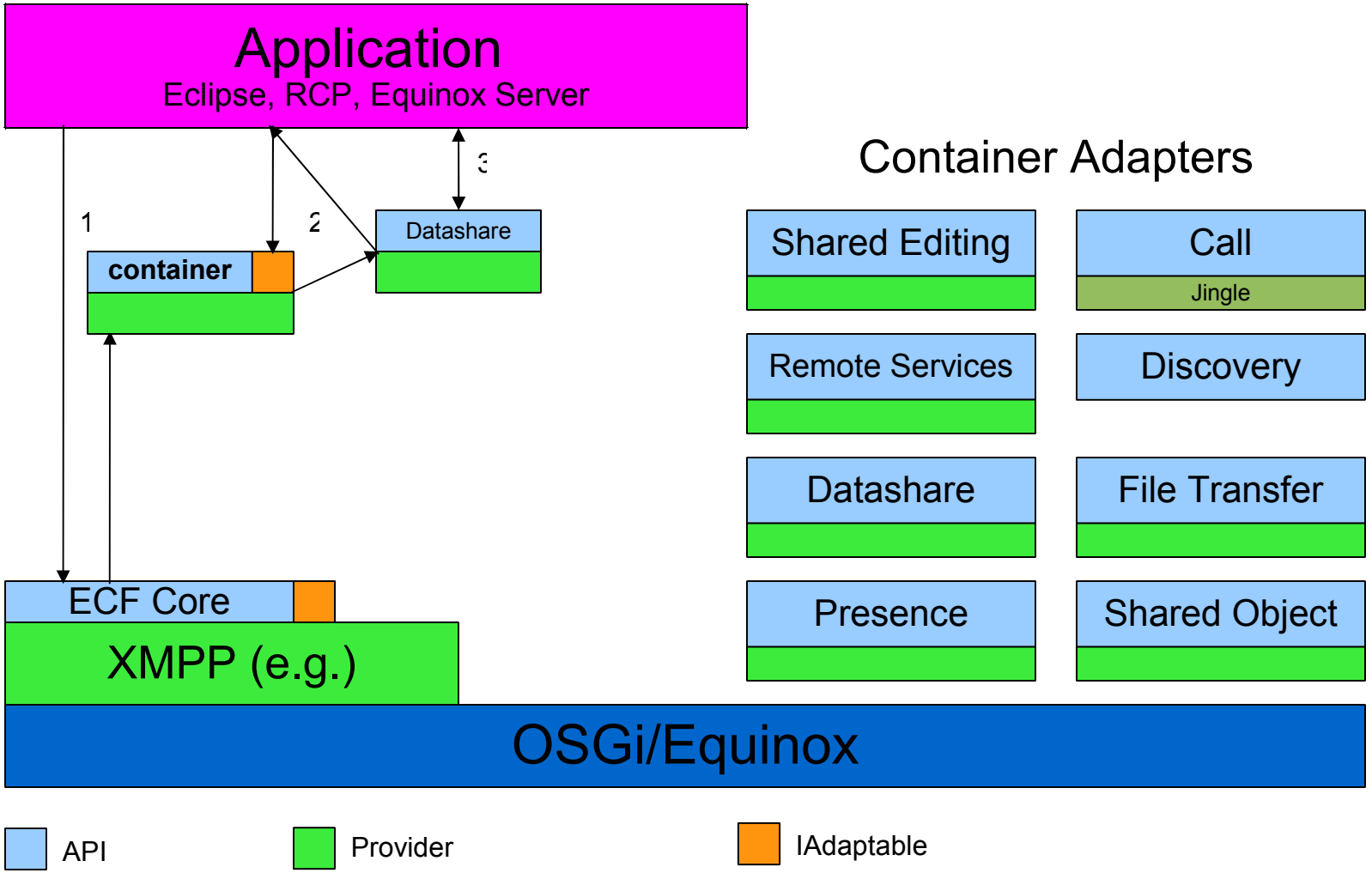
# Can Absence Make a Team Grow Stronger?

- Rule 1: Exploit Diversity
- Rule 2: Use Technology to Simulate Reality
  - Virtual Spaces for Team and Community
- Rule 3: Hold Team Together:  Build Trust
  - Team
  - Community

# Extensibility

- APIs through Adapters
  - Core:  2 bundles ~120k
  - API plugins/bundles – aka Container Adapters
- Providers Implement:  core + 0 or more Adapters
- Extensibility:  Exploits Diversity In Community
  - Shallow or Deep Additions
  - Sense of Ownership

# ECF Provider Architecture



# Code

```
 IContainer container =
ContainerFactory.getDefault().createContainer();

 IChannelContainerAdapter adapter =
(IChannelContainerAdapter)
container.getAdapter(IChannelContainerAdapter.clas
s);

 IChannel channel = adapter.createChannel(...)
```

# Asynchronous File Transfer

- org.eclipse.ecf.filetransfer
  - API
    - Asynchronous notifications to listener
      - Start,Data,Done
  - Providers
    - URLConnection (JRE)
    - Apache httpclient 3.0.1,
    - SCP/SSH
    - EFS
  - APIs:  receive, send, browse
- Now being used for p2 and SDK

# Demo – Asynchronous File Transfer

# Remote Services

- org.eclipse.ecf.remoteservice
  - API
    - **Clients have choice**
      - **Proxy**
      - **IRemoteService**
        - Asynchronous (Listener) Invocation
        - Futures
  - Providers
    - R-OSGi
    - JMS - ActiveMQ/BEA
    - ECF generic
    - JavaGroups
    - Riena
    - Others (?)

# Demo – Remote Environment Info Service

# ECF Team Communications

## Team Communications

- Build Diversity, Trust, and Community
- Technology Aids
  - Interoperability
  - Integration
  - Extensibility
- Project: http://www.eclipse.org/ecf
- Wiki:  http://wiki.eclipse.org/Eclipse_Communication_Framework_Project
- IRC:  irc://irc.freenode.net/eclipse-ecf