

# Using ECF for Lightweight Distributed Team Collaboration

Scott Lewis - ECF Project Lead

Markus Kuppe – ECF Committer

<http://www.eclipse.org/ecf>

Teamwork: A few harmless flakes working together can release an avalanche of destruction



## Teams they are a changin'

- Membership
  - ♦ Distributed
  - ♦ Cross-organizational/Multi-organizational
  - ♦ More Diverse Skills, Backgrounds, Cultures
- Project Organization
  - ♦ Flat(ter)
  - ♦ Voluntary
  - ♦ Members Have More Control
- Community Expectations
  - ♦ Open Planning
  - ♦ Responsiveness
    - Bug reports
    - Contributions
    - Newsgroups, Mailing Lists, IRC, etc.

# Technology for Today's Teams

- Interoperability
  - ♦ Multiple protocols, multiple clients, multiple services
- Integration
  - ♦ Reduce complexity
- Extensibility
  - ♦ Customizable
    - Processes and conventions
  - ♦ Team-specific Tools
  - ♦ Dynamic discovery of services between peers

## 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 via Communication

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

# Can Absence Make a Team Grow Stronger?

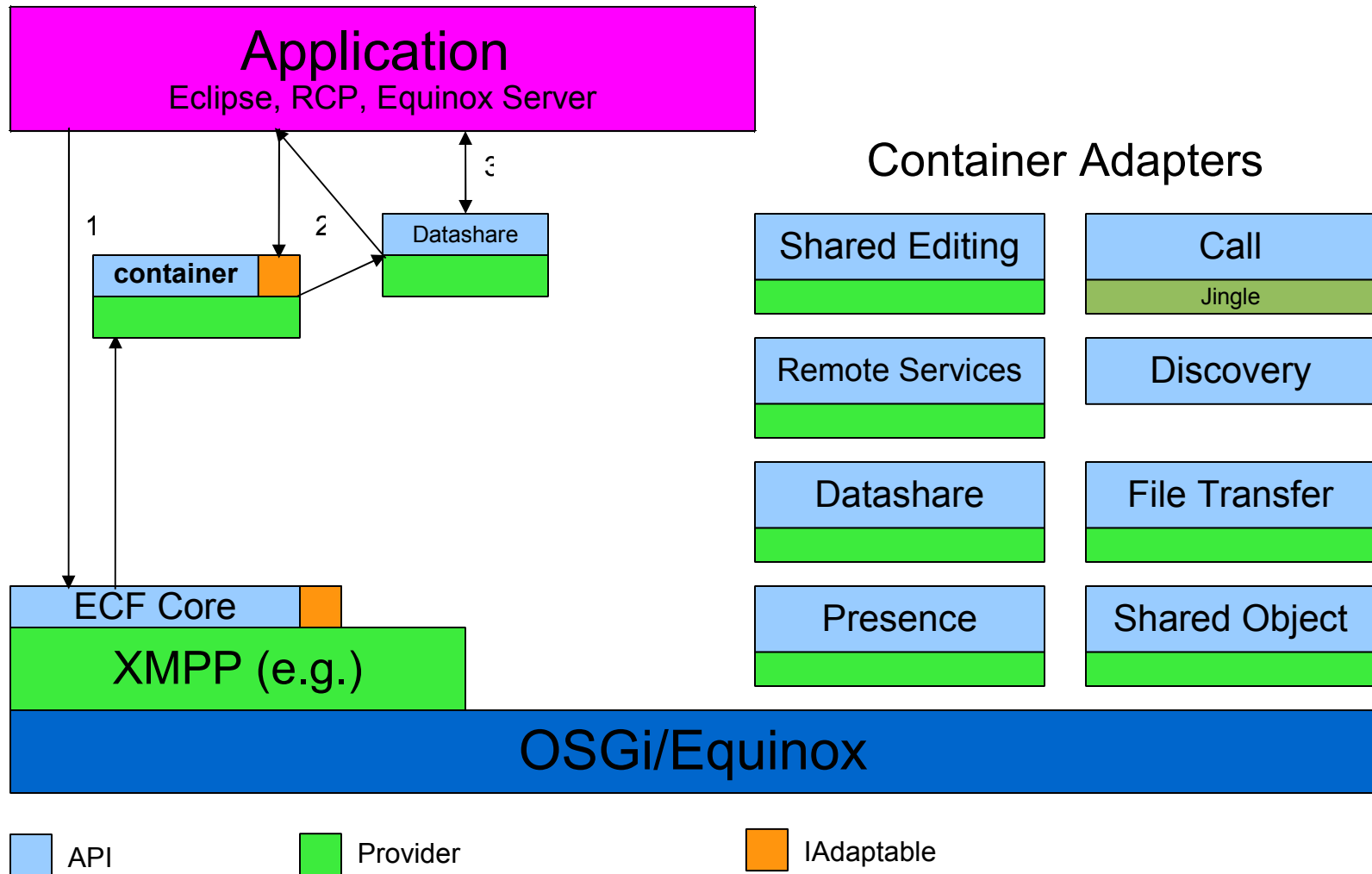
Ann Majchrzak, et. al.

- Rule 1: Exploit Diversity
- Rule 2: Use Technology to Simulate Reality
  - ♦ Virtual Spaces for Team and Community
- Rule 3: Build Trust
  - ♦ Member-to-Member
  - ♦ Member<->Community



# Demos

## ECF Provider Architecture



## Code

```
IContainer container =  
    ContainerFactory.getDefault().createContainer();  
  
IChannelContainerAdapter adapter = (IChannelContainerAdapter)  
    container.getAdapter(IChannelContainerAdapter.class);  
  
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
    - Bittorrent
    - EFS
  - ♦ Receive, Send, Browse
- Used for p2

## 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 (?)

# 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](http://wiki.eclipse.org/Eclipse_Communication_Framework_Project)
- ◆ IRC: <irc://irc.freenode.net/eclipse-ecf>