Open Systems:
Open Source for the Rest of Us

Eric Scharff
Design, Learning and Collaboration
22-Apr-2002
Overview

- Open Systems
- Open Source
- Open Source + Open Systems
Open Systems

• What are Open (Evolvable) Systems?
  – Collaboration and evolution
  – Open Source and Open Evolvable Systems

• Understanding Existing Open Systems

• Open Evolvable Systems Challenges
  – Supporting Change
  – Coping with change

• Why Open Systems?
The Access Mindset
The Access Mindset
Collaborative Construction
Ask Jeeves - Get an answer
Experts Exchange - Topics
Experts Exchange - Questions
From: cheauwei  
Title: "Printing a graph in pop-up dialog box"

Status: Waiting for answer.  
Date: Tuesday, February 22 2000 - 07:38AM MST

Points: 500

I had created an applet with a "Draw Graph" button. When user push on the button, a frame pop-up. In the pop-up frame, they can plot and draw a graph. My problem is i want to have a print button in the pop-up frame so that user can print out the graph they drawn.

Please send me the source code. Thanks a lot.

From: diakov  
Date: Tuesday, February 22 2000 - 08:08AM MST

You can try printing your swing based component using the utility class as provided in this:

```java
import java.awt.*;
import javax.swing.*;
import java.awt.print.*;
```
Closed and Open Systems

- **Ask Jeeves**
  - Professional ‘experts’ answer questions
  - Staff chooses what questions to answer
  - No understanding of quality of answer
  - Unanswered questions are “invisible”

- **Expert’s Exchange**
  - Many people contribute knowledge
  - People contribute answers to posed questions
  - Answer assigned “points” by questioner
  - Community notified if no answer exists

- **Closed Evolution**
  Controlled by Staff

- **Open Evolution**
  Controlled by Users
Open Evolvable Systems

- Open Systems are processes in which change is possible “at all levels.”
  - Processes change
  - Change is a “first-class activity”
- Easiest to contrast with closed systems, where boundaries are fundamentally fixed.
- Important for wicked and inherently collaborative problems
  - Socio-technical problem
What is Open Source?

- A software engineering technique?
- A mindset?
- A culture?
- A religion?
Open Source is Misunderstood

YOU WANT TO WHAT?!

INSTALL LINUX ON THE iFRUIT.

ABSOLUTELY NOT! I LIKE MY COMPUTER THE WAY IT IS!

I CAN PARTITION IT. YOU WON’T EVEN NOTICE.

HOW CAN ANYONE NOT NOTICE, JASON?!

OPEN SORES! YUCK!

SOURCE, MOM. OPEN SOURCE.
Open source is NOT

- Zero cost software
- Zero cost labor
- Completely decentralized
- Lacking fixed goals
- A “hacker phenomenon”
Perspectives on Open Source

- Software design, intellectual property, high quality software dominant perspectives
- Open source is this and more…
  - “Given enough eyeballs, all bugs are shallow” (Raymond, 1998)
  - “Free software is a matter of liberty, not price” (Stallman, 1992)
  - “Open source folks tend to build the tools they need or wish they had” (Vixie, 2000)
  - “Now, this talk is supposed to be about the Culture of Perl. Some would say “What Culture?” To which the only adequate response is, “Well, given a suitable medium, even bacteria are allowed to have a culture.”” (Wall, 1997)
Collaborative Construction

- Open source software communities provide examples of ongoing activities where people understand and solve problems of mutual interest.
  - Social *and* technical process
  - Complex interrelationships
  - Continuous evolution
- Collaborative construction facilitates application outside of software design
Case Studies

- SourceForge
- Python
- Perl

Feather

Classic.com

Slashdot
News for Nerds. Stuff that matters.
Case Studies (2)
Open Source Framework

- Definable Project
- Exciting Project
- Project Leader
- Work Assignments
- Participants

- Standard Distribution
- Project Framework
- Collaborative Technology
- “Wish List” or TODO list
Participants

- Active contributors through creation, use, and refinement of the system
  - continuum of involvement
  - inclusive process (?)
- Motivation Challenge (why do it?)
  - personal interest / utility
  - social capital
- Course design
  - mix of intrinsic and extrinsic motivators
  - personally meaningful projects
Flow

- 63% - This project is as creative as anything I have done
- 72% - When I program, I lose track of time
- 59% - With one more hour in the day, I would spend it programming
- 48% - Like composing poetry or music
Collaborative Process

- **Social conventions emerge governing communication and change adoption**
  - encourage participation
  - creation, sharing, adoption of changes
- **Leadership Challenge (who does it?)**
  - leadership emerges naturally from activity
  - open participation versus leader control
  - leadership is hard
- **Course design**
  - project team leadership emerged
  - initial project ideas from my “wish list”
Collaborative Technology

• Tools used to communicate and coordinate development
  - communication - email, chat
  - coordination - CVS

• Coordination Challenge (how is it done?)
  - managing heavy traffic
  - duplication of effort

• Course design
  - Swiki - open collaborative Web site building
  - overcome “media competition”
Object Produced

• Concrete artifacts that comprise the software system
  – software system used by participants
  – non-software objects (examples, documentation)

• Extension Challenge (how does it change)
  – systems need to grow through change
  – competency to make changes required

• Course design
  – EDC extension
  – Swiki contribution
Open Source & Open Systems

• What happens when the object produced isn’t a software system?
• “Open Source for the Rest of Us”
  - non-technical problem domains
  - user communities not entirely software builders
• Aspects of open systems
  - provide opportunities for extension and modification as part of their use
  - grow through this extension process
  - support users needs to collaboratively construct
  - facilitate open (inclusive) participation in open (ended) domains
EDC Redesign

• Motivation Challenge
  - new domain models and participants
• Leadership Challenge
  - allow leadership to emerge
• Coordination Challenge
  - utilize Swiki technology
• Extension Challenge
  - improved physical object recognition and software infrastructure
Framework Adoption Challenges

- **Finding common ground**
  - Acknowledge “symmetry of ignorance”

- **Getting users to participate**
  - Tasks not perceived as collaborative
    - Handle real problems shared by people
  - Motivation (lack of time / inclination)
    - Be as inclusive as possible
  - Lack of technical sophistication
    - Provide many avenues

- **Handling change**
  - Evolution a result of change in context