

Transit and Open Source: Is it an Option?

Paula Okunieff

October 16, 2005

Topics

- Terms and Definitions
- Open Source Business Models
 - Impacts
 - Benefits
- Is OSS an Option?
 - Criteria for Success
- Possible Strategies to Transit OSS

Terms

- Open System
 - Open Architecture
 - Open Standard
 - Open Interface
 - Open Source Software
-
- “Open” – accessible to all, transparent, freely available to all

Open System Defined

- The Department of Defense defines open system as:
- “A system that implements *sufficient* [sic] open specifications for interfaces, services, and supporting formats to enable properly engineered components to be utilized across a wide range of systems with minimal changes, to interoperate with other components on local and remote systems, and to interact with users in a style that facilitates portability.”

Open Architecture Defined

- “**Open architecture** is a type of computer architecture that allows users to upgrade their hardware in all of the computer components (for example the IBM PC has an open architecture).”
- From wikipedia.org

Open Standard Defined

- Standard related to a specific domain that is developed and promulgated in a transparent, consensus based environment by industry experts.

Open Interface

- A public specification of those properties of a software component that other components and software may rely upon to access functions and data internal to the component.
- Adapted from wikipedia.org

Open Source Software Defined

- Software that is associated with a class of license agreements that follow the requirements described in the Open Source definition (see opensource.org).
 - Freedom to redistribute source code and executable
 - Freedom to change source code
 - Cannot restrict use or redistribution of original or altered code

Who develops open source? And how is it developed?

- Motivated programmers
 - Self selected
 - Project Administrator(s)
- Supported throughout Lifecycle using Collaboration Tools
 - Mailing lists
 - Roadmap, patches, enhancements
 - Wiki boards (for coding and collaborating)
 - Bug Tracking and configuration management
 - Infrastructure tools (e.g., compilers)
 - Concurrent version systems (CVS)

Open Source Development Models

- Commercial
 - May be “Loss Leader” to compete against formidable market obstacles
 - Believe in OSS
- “Ad-Hoc”
 - Informal group of developers working on OSS development web sites
- Community Source
 - Project level
 - Non-profit corporation, governance structure

Open Source Development Models

- Foundations
 - e.g., Apache Software Foundation
 - Family of products
- Repositories (shared software)
 - e.g., Government Open Code Consortium
 - Membership based

OSS Model Features: Commercial

- E.g., Mozilla
- Vendor controls official releases
- Requires contributors to sign copyright / patent release
- Provides warranty and support services
 - Does not exclude others from providing similar services

OSS Model Features: “Ad Hoc”

- E.g., Linux, Map Builder
- Self-appointed key project administrators who “manage” and make decisions on code
- Self-selected, motivated programmers working from a “roadmap”
- Use public collaborative web site (sourceforge.net)
- IP owned by each contributor

OSS Model Features: Community Source

- (see handout by Lois Brooks)
- Resource commitment from “community” organizations
 - for development, maintenance and upgrade
- Governance and legal structure (non-profit)
- Appointed project manager(s)
 - Application of structured, software engineering approach

OSS Model Features: Community Source, cont.

- Dedicated programming resources
- May use collaborative web services or equivalent
- May offer services and training support
- IP owned by non-profit

OSS Model Features: Foundation

- E.g., Apache Foundation
- Formal policies and facilities to ensure Interoperable Applications and official versions
- Governance rules and legal structure
- Use private collaborative sites for development

OSS Model Features: Foundation, cont.

- Corporate and individual resources
 - Vet developers for skills, competence and availability
- Develop incubator for prototyping applications
- IP owned by Foundation

OSS Model Features: Repository

- E.g., Government Open Code Collaborative (GOCC)
- Repository to share code from members
- No formal governance structure
- Legal organization (non-profit)
- IP owned by Organization posting application

OSS Benefits

- Ability to choose among technical support contractors to implement and maintain the software
- Ability to customize the code to meet specific requirements
- Flexibility in customizing application to meet requirements

OSS Benefits, cont.

- Software support and maintenance may be extended indefinitely
- Ability to scale application to meet specific performance criteria

OSS Impacts

- Fragmentation or code “forking”
- Lack of compatible/interoperable applications
 - Mitigated with open standards and reference data model
- Version control
- Building a viable community of programmers
- Contract/train staff for support and service
- Competition from proprietary vendors

OSS Issues for Transit

- What transit applications may work as OSS?
- What OSS business model(s) best supports transit procurement policies?
 - Who should develop the transit OSS?
- What OSS applications may carriers/regions deploy first?

What will make Transit OSS products successful? Critical Success Factors

- Education and participation by transit agencies
- Ensure interoperable suite of software tools
- Development of robust, maintainable code through best practices software development processes

Critical Success Factors, cont.

- Development of a support and warranty service sector
- Support a dedicated community of programmers competent in transit software
- Legal protection for contributors and users of OSS

Business Models that Meet CSFs

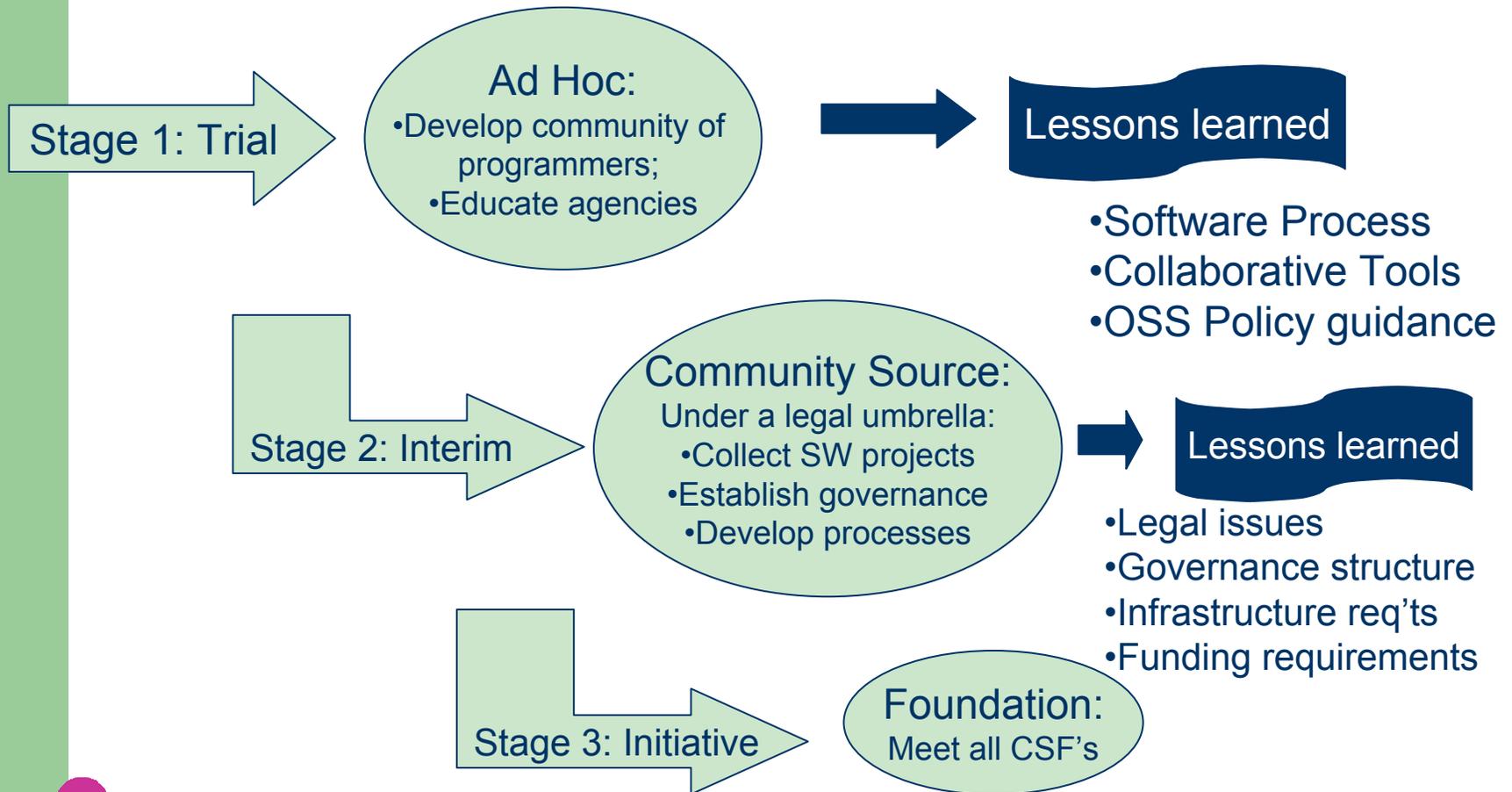
	Ad Hoc	Found-ation	Comm. Source	Repos-itory
Interoperable SW		✓		
SW Dev.Process	✓	✓	✓	
Service Sector	✓	✓	✓	
Community	✓	✓	✓	
Legal Umbrella		✓	✓	✓

Transit and Open Source: Is it an Option?

There are clear opportunities for transit in the area of OSS

- but given the newness of the topic to transit, we need:
 - education and outreach
 - lessons learned
 - a proof-of-concept

Potential Strategies to Transit OSS



Thank you



Questions