Open-Source Business Models:
Making money by giving it away

Andrew J. Hall
HALL LAW

February 20, 2015
Linux Collaboration Summit
Santa Rosa, CA

© 2015 Hall Law. All rights reserved. This presentation is licensed for use and distribution under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 (CC BY-NC-ND 4.0) available at: https://creativecommons.org/licenses/by-nc-nd/4.0/legalcode
Presentation Overview

› Open-Source Business Models
› Open-Source Development
Open-Source Business Models
Open-source Business Models

Open-source business models usually rely upon one or more of the following strategies:

1. **Dual-licensing** proprietary company software;
2. Providing **commercial** or **enterprise** versions, **plugins**, or **extensions** to open-source products;
3. Offering **maintenance**, **support**, **consulting**, or other **services** that **support** or **complement** open-source products;
4. Offering **hosting**, **warranty**, **indemnity**, or other **products** that **complement** open-source products; and
5. **Closed-source modified distributions** of open-source products.
## Open-Source License Types

<table>
<thead>
<tr>
<th>License Type</th>
<th>Intended Copyleft Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissive (Apache-2.0, BSD)</td>
<td>None</td>
</tr>
<tr>
<td>Weak Copyleft (LGPL, MPL, CDDL)</td>
<td>Modifications/enhancements to the open-source software</td>
</tr>
<tr>
<td>Strong Copyleft (GPL, AGPL, OSL)</td>
<td>Certain software distributed in combination with the open-source software.</td>
</tr>
<tr>
<td>Network Strong Copyleft (AGPL, OSL-3.0)</td>
<td>Certain software distributed or hosted in combination with the open-source software.</td>
</tr>
<tr>
<td>Prohibitive (Ms-LPL, BCLA)</td>
<td>Typically none, but specific uses (e.g., commercial) are prohibited</td>
</tr>
</tbody>
</table>
1. Dual Licensing (1/2)

› Company offers its own proprietary software for use under either of (a) an open-source license or (b) a paid commercial license.

› The open-source license often prohibits or limits commercial use of the software. Licensees wishing to avoid such prohibitions and limitations can purchase a commercial license with more favorable terms.

› Commercial licenses may additionally or alternatively:
  (a) provide access to company services (e.g., support, maintenance, and customization);
  (b) include warranties or indemnification not available under the open-source license;
  (c) provide early access to updated versions of the software; or
  (d) resolve infringement allegations made by the licensor.
1. Dual Licensing (2/2)

› **Examples:**
  MySQL, BerkeleyDB, Java EE/SE, MongoDB, Asterisk, Modelio, iText, Wurfl, Qt.

› **Challenges:**
  – Third parties may fork the software creating alternative implementations (e.g., MySQL forks MariaDB and Drizzle)
  – Collecting fees may require unpopular licensing campaigns targeting the software’s user base.
2. Open Core & Open Platform (1/1)

› **Open Core (Freemium):** Company offers a version of its product under an open-source license while offering enhanced or “enterprise” versions of the software under a commercial license.

› **Examples:** Sendmail, Java EE/SE, Sourcefire Snort, Qt, SugarCRM

› **Challenges:**
  - Potential for forking and third-party premium extensions
  - Limiting access to premium versions can negatively impact community response and adoption
2. Open Core & Open Platform (2/2)

› **Open Platform**: Company releases a platform or other software under an open-source license and offers proprietary plug-ins, extensions, applications, or content under commercial licensing terms through the platform.

› **Examples**: Android, Eclipse, Hadoop, Wordpress

› **Challenge**: Success often particularly dependent on broad platform adoption and sometimes third-party participation.
3. Providing Complementary Services

- Company offers services that relate to or support customer use of open-source software; software may or may not also be licensed by the company.
- Related services include, for example, training, customization, implementation, maintenance, certification, and support services.
- Selling additional services typically require the company investment of significant additional human or technology resources.
- **Examples:** Red Hat, Canonical, Novell, Pivotal, Palamida, Black Duck.
- **Challenges:**
  - Third parties can typically provide competitive services with limited barriers to entry
  - Service models typically don’t scale well
4. Providing Complementary Products

› Company offers products that complement or heavily depend upon the use of open-source software.
› Related products include, for example, hosting services, hardware and virtual systems incorporating open-source operating systems and platforms, and providing warranties and indemnification for open-source software and use.
› Additional product can typically be sold without significant additional resource investment from the providing company.
› **Examples:** Red Hat’s “IP Assurance Program” and RHEL build services; MongoDB and AWS hosting services.
› **Challenges:**
  - Third parties can typically provide competitive products with limited barriers to entry
5. Closed-Source Distributions

› Company releases commercial (closed-source) versions of open-source originally licensed under a permissive license (e.g., Apache 2.0) or offers commercial plugins or extensions to an open-source project or platform. The distributions are often specialized for a particular industry or use case.

› **Examples:** Cloudera, Hortonworks, MapR and AWS (offering commercial versions of the Apache Hadoop project).

› **Challenges:**
  - Third parties can offer competitive services;
  - Success of commercial versions, plugins, and extensions may depend on the success of the underlying software or platform
Additional Open Source Revenue Models:

› **Donation-based funding:** covering expenses through contributions to the project or a related foundation.

› **Open-source bounties:** companies or communities offering bounties for bug and security fixes or functional improvements

› **Crowdfunding:** individual, groups, or companies offer to implement functionality for general release under an open-source license in exchange for a set fee.

› **Branded merchandising** such as merchandise sold by the Mozilla and Wikimedia Foundations.

› **Ad-Supported Software**
### Apache Hadoop Governance

**Apache Software Foundation (ASF) Board of Directors**

- Manages ASF assets (funds, IP, equipment)
- Allocates ASF resources to projects
- Appoints PMC Chairs

**Key**

- Appointed by Board
- Appointed by PMC
- Elected by community

**Apache Hadoop Project Management Committee (PMC)**

- Appointed by the Board from among PMC Members (also an ASF Officer as VP, Hadoop)
- Rotates annually among PMC Members; PMC Members make recommendation to Board

- Approve (by consensus) and remove (by majority) PMC Members
- Approve (by consensus) and remove (by majority) Committers
- Approve (by majority) new Hadoop releases
- Revise (by majority) project Bylaws

**Committers**

- For each release, the Release Manager is a volunteer from among Committers
- Responsible for building consensus on release content

- Appointed from among Contributors with consensus approval of PMC
- Have access to the project codebase and may make changes with consensus approval of Committers.
- Must sign Contributor License Agreement (CLA)

**Contributors**

- Submit bug reports, code changes, and documentation
- Contributors that make sustained contributions may be invited to become Committers
Open Development (2/2)

Android Governance

Key:
- Google Employee
- Community Member

- Senior Google engineer overseeing development of a sub-project
- designate Approvers and Verifiers

- Project Leads
  - Designates Approvers and Verifiers

- Approver
  - Approves or rejects code changes

- Verifier
  - Reviews and tests contributed code

- Individual Contributors
  - Must sign Individual Contributor License Grant

- Non-Google Corp. Contributors
  - Must sign:
    (i) Corporate Contributor License Grant
    (ii) Individual Contributor License Grant

- Google Employee Contributors
  - Have the same access to code base as community contributors.
Questions, Comments, Thoughts?

Contact:
Andrew J. Hall
ahall@thehalllaw.com
www.thehalllaw.com