The Business Model of Commercial Open Source Software

(Or: How MySQL AB Got to a $1BN Valuation)

Dirk Riehle
SAP Research, SAP Labs LLC

dirk@riehle.org - http://dirkriehle.com - http://twitter.com/dirkriehle

Commercial Open Source

- Commercial open source definition:

> "Commercial open source is [open source] software that a for-profit entity owns and develops. The company maintains the rights and determines what is accepted into the code base [...]" [1]

- Core product is available to community under open source license
  - Available in source code form, modifiable, redistributable
  - See http://www.opensource.org for "approved" licenses

Growth of Commercial Open Source

- Gartner Group prediction:

"By 2012, at least 50% of direct commercial revenue attributed to open-source products or services will come from projects under a single vendor's patronage." [2]

- Open source is growing exponentially, on a non-trivial base [3]
  - Individual projects are growing linearly or polynomially
  - **Number of viable projects is growing exponentially**
  - Projects are growing in every conceivable domain

Benefits over Traditional Commercial Software

All other things being equal
commercial open source firms

can **go to market faster**

with a **superior product**

at **lower operations cost**

and **sell more easily**

than possible for traditional competitors [4].

A Simple Open Source Project Classification

	single proprietor	community-owned
single product or product line	**Commercial Open Source** (e.g. MySQL)	Community Open Source (e.g. PostgreSQL)
multi-product assembly ("stack")	Commercial Distribution (e.g. Red Hat)	Community Distribution (e.g. Debian)

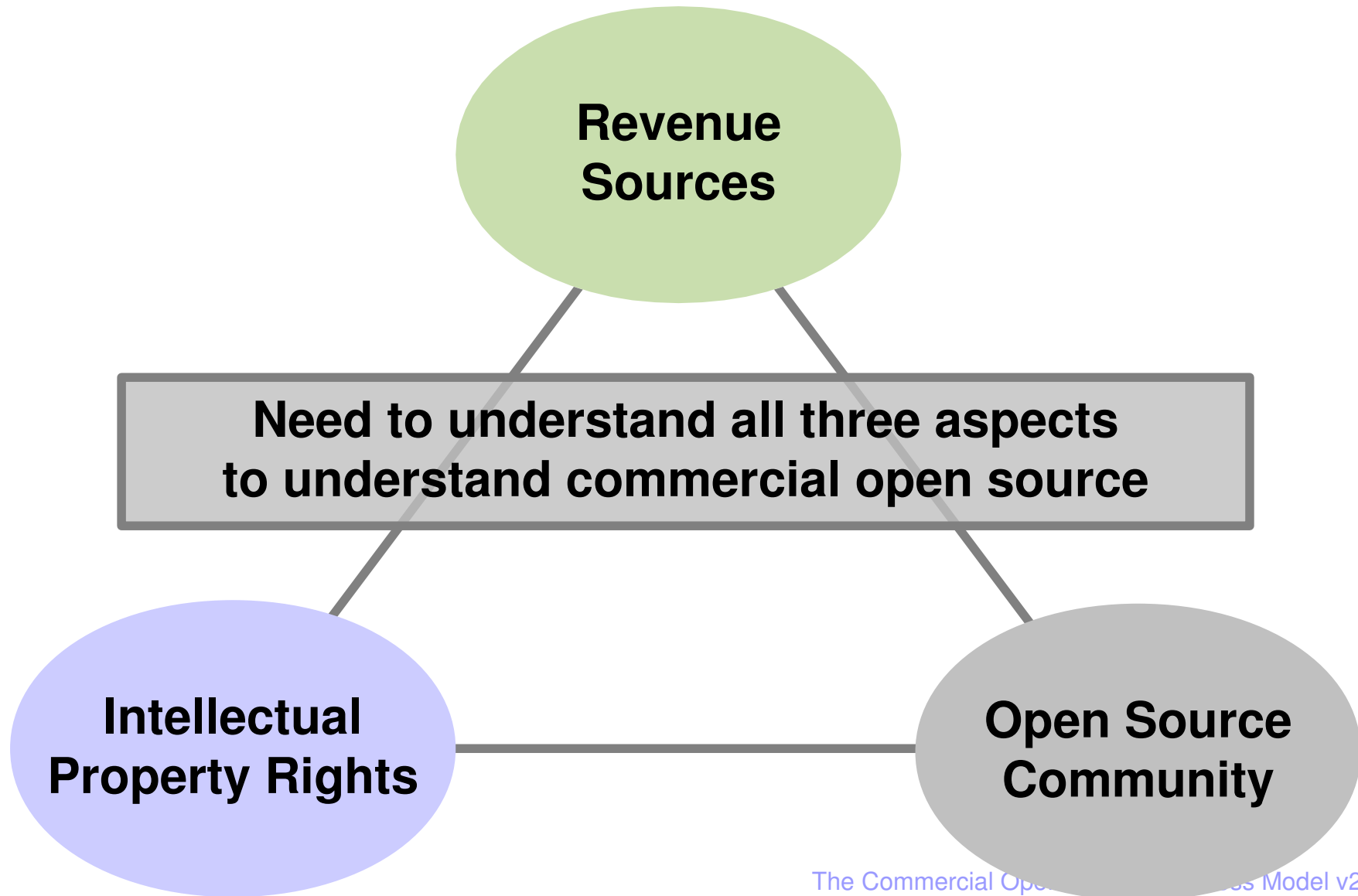# Commercial vs. Community Open Source [1]

Commercial Open Source

- One single proprietor
- Typically dual-licensed
- Increasingly open-core
- Venture capital backed
- Significant direct revenues
- Asymmetric community

Community Open Source

- Community of owners
- Single license
- No functionality withheld
- Cross-subsidized
- No direct revenues
- Symmetric community

What's Different in Commercial Open Source
(When Compared With Traditional Closed Source Firms)?

Revenue Sources

**Need to understand all three aspects
to understand commercial open source**

Intellectual Property Rights

Open Source Community

Revenue Sources of Commercial Open Source

- ## Core product
  - Provision of the software under a commercial license
  - Formerly equated with dual-license strategy

- ## Whole product
  - Provision of an extend version of the software under a commercial license
  - Now being called "open core" model

- ## Operational comfort
  - Provision of a quality-controlled update and bug-fixes service (subscription model)
  - Hotline support, bug-fixes on demand, 24x7 availability

- ## Support, consulting, and implementation services
  - Consulting and implementation services to customers and resellers
  - Second level support for OEMs and resellers

The Intellectual Property Rights Imperative

Goal: Provide product as open source to community but stall any possible commercial competition.

- Using a reciprocal ("viral") license like the GPL stalls competitors
  - GPL prevents closed extensions that build competitive advantage
 - Puts everyone on equal footing **except for the founding company**

Intellectual property rights imperative: Ensure that this firm and only this firm has the relicensing rights for the software.

- Tightly control contributions to code base; require rights grant

Some Benefits of a User Community

- For sales: Lead is already using core product; in-house champion may help sales process

- For marketing: Product is widely known, hopefully loved, and supported by community

- For product management: Feedback is fast and comprehensive, getting users close to firm

- For engineering: Feedback improves product, contributions speed up development; potential employees are pre-screened

- For support: Self-supporting community lowers overall support costs, makes documentation more comprehensive

Average Structure of User Community

Non-paying Users
(50X-500X, 0%)

Casual Customers
(20X, 2%)

Enterprise
Customers
(1X, 98%)
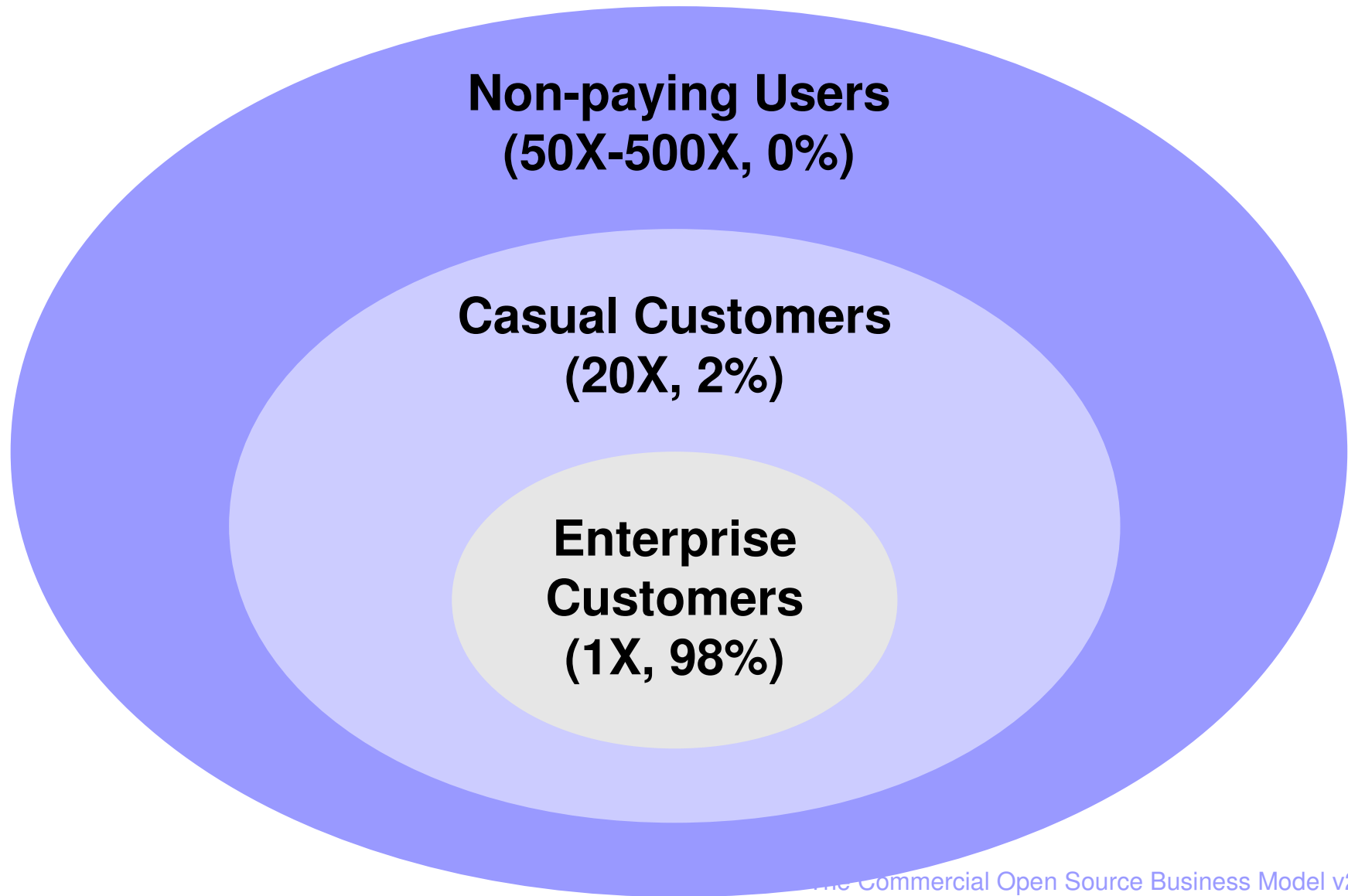
Business Function: Community Management

Goal: Drive adoption of free core product by growing a self-supporting open source community.

- Lead community to deploy and use core product
- Lead community to evangelize more new members
- Lead community to be self-supporting wrt core product
- Firm can steer its investment by user-conversion metric
- Many best practices of community building apply

The Sales Funnel of Commercial Open Source [5]
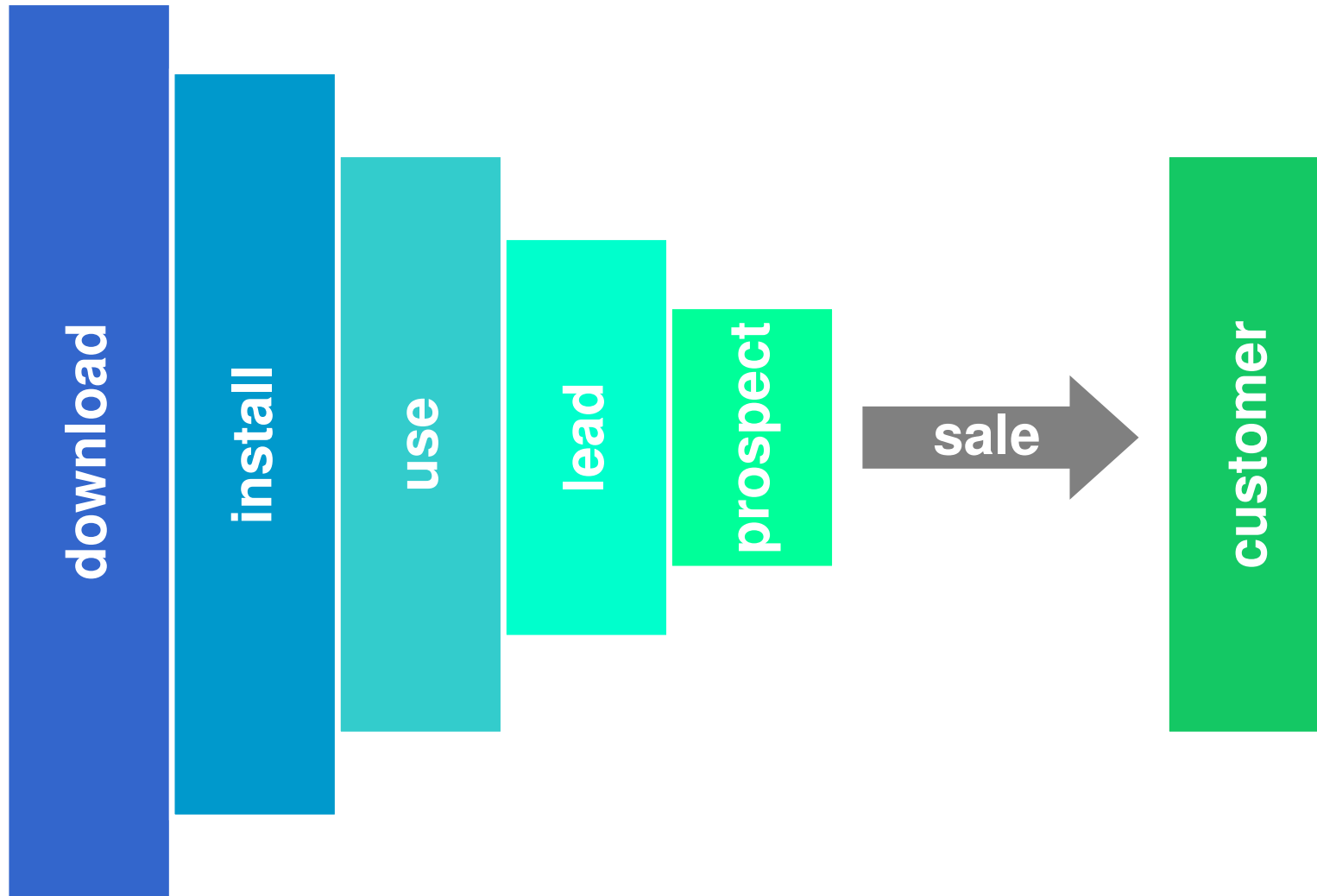
Business Function: Sales

Open Source Sales

- Let the numbers play for you
- Reduce friction to the max
- Call on in-house champion
- Leave user community alone
- Many more best practices...

Traditional Sales

- Traditional sales funnel
- Luke-warm or cold calls
- Other well-known practices...

**MySQL frequently finds in-house champion
in competitive sales situation [6].**

Community makes sales faster and easier.

Business Function: Marketing

- Much of commercial open source marketing is traditional
- User community evangelizes, micro-markets, virally
- Great and early source of believable testimonials

**MySQL has found that its user community
helps spread the word and evangelize new members [7].**

Community makes marketing cheaper and more effective.

Business Function: Product Management

- User community provides fast and comprehensive feedback
  - Feedback on functional needs and issues (e.g. usability)
  - Faster and more immediate and more accessible than in traditional setting
  - Helps prioritization and triage of features and road maps
- User community is source of user innovation
  - Source of new feature ideas and insights
  - Helps find new application ideas
  - May even do biz dev for firm

MySQL got rapid feedback from its user community that was innovative and helped focus the product [8].

Community gets product management closer to user.

Business Function: Engineering

- User community provides fast and comprehensive feedback
  - Fast feedback on functional and non-functional issues (bugs, performance, …)
  - Also for most unusual circumstances

- User community extends core solution into new applications
  - Users can help themselves and develop free open source add-ons
  - Effectively, free market research and exploration
  - Pre-screening of potential future employees

MySQL got rapid feedback about bugs in the software and was able to identify and hire top developers [6].

Community makes engineering faster, better, cheaper.

Business Function: Support

- The user community is self-sustaining
  - Implies its own extended documentation
  - Fast source of support at any time of the day
  - Web-searchable content frequently faster than hotline

- All of this reduced support costs

The dev.mysql.com website is full of user-generated content that helps users and customers alike solve problems.

Community makes support cheaper.

Conclusions and Summary

All other things being equal, commercial open source firms can **go to market faster** with **a superior product** at **lower operations cost** and **sell more easily** than possible for traditional competitors.

- Distinguish between commercial and community open source

- To be successful, commercial open source software firms need to
  - Adjust to **new or modified new revenue sources**
  - Follow **the Intellectual Property Rights Imperative**
  - Develop **new skills of community management**

- Claims are validated by review of primary sources

References

[1] Dirk Riehle. "The Economic Motivation of Open Source." IEEE Computer April 2007. Page 25-32.

[2] Mark Driver. "Predicts 2009: The Evolving Open-Source Software Model." Gartner Group, Inc. 2008.

[3] Amit Deshpande, Dirk Riehle. "The Total Growth of Open Source." In Proc. of OSS 2008. Springer, 2008.

[4] Dirk Riehle. "The Commercial Open Source Business Model." In Proc. of AMCIS 2009. AIS, 2009.

[5] Larry Augustin. "Smoothing the On-Ramp to Commercial." Presentation given at OSBC 2008.

[6] MIT. "An Interview with Marten Mickos." MIT Sloan Management Review (Fall 2008). Page 15pp.

[7] Christof Wittig and Sami Inkinen. "MySQL Open Source Database in 2004." Stanford GSB Case SM-124. Stanford, 2004.

[8] The Wallstreet Journal Online. "Software Firm is Open for Innovation." WSJ: July 7, 2008.

Document License: CC-BY-SA 3.0

This document is licensed under the Creative Commons license BY-SA 3.0, see http://creativecommons.org/licenses/by-sa/3.0/

The document's original author and copyright holder is Dirk Riehle. The author can be reached at dirk@riehle.org. The document can be found at http://dirkriehle.com.

You are allowed to share and create derivatives of this document, even for commercial purposes, but you must keep this attribution page included in the document and you must show it as part of your performance. The title slide must visibly bear the following two lines of text:

You may add your own copyright notice for the portions of the document you added.

Thank you! Questions?

Feedback welcome!

dirk@riehle.org - http://dirkriehle.com - http://twitter.com/dirkriehle