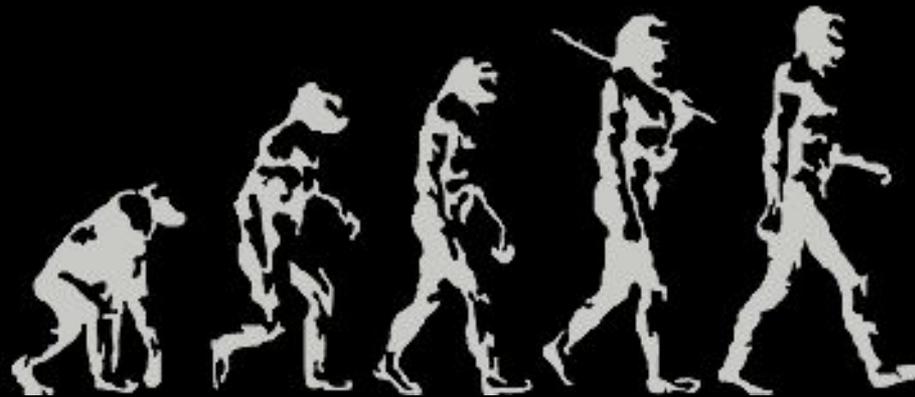


From support services to software services

The evolution of open source business strategies



Matthew Aslett, Senior Analyst,
Enterprise Software, The 451 Group



Open source is not a business model

How do vendors generate revenue from open source software?

Published:
October 2008

Enquiries to:
sales@the451group.com

the 451 group
Analyzing the Business of Enterprise IT Innovation

OPEN SOURCE IS NOT A BUSINESS MODEL

How Vendors Generate Revenue from Open Source Software

Open source software is freely available, but vendor lock-in comes at a price. This report assesses the development, licensing and revenue-generation strategies used by vendors to make money from open source software.

CAOS COMMERCIAL ADOPTION OF OPEN SOURCE

4 FINDINGS	5 IMPLICATIONS	1 BOTTOM LINE
<ul style="list-style-type: none">• The majority of vendors analyzed utilize some form of commercial licensing to distribute, or generate revenue from, open source software. PAGE 10• Half of the vendors assessed in this report are combining code developed via open source projects with software developed out-of-sight of open source project members. PAGE 11• Vendors using hybrid development and licensing models are balancing higher development and marketing costs with the ability to increase revenue-generation opportunities from commercially licensed software. PAGE 20• The license used for an open source project (technical or permissive) has a strong influence on development, vendor licensing and revenue-generation strategies. PAGE 20	<ul style="list-style-type: none">• The line between closed and open source has blurred as FOSS is embedded in proprietary products and commercial extensions have been added to FOSS. PAGE 07• Vendors must choose complementary development, licensing and revenue strategies in order to maximize revenue-generation opportunities. PAGE 20• Customers make sure that they are aware of vendors' strategies so they can understand and predict the behavior of vendors encouraging them to become paying customers. PAGE 07• Increased use of commercial licensing could create tension between the expectations of potential customers and commercial reality. PAGE 06• Vendors need to consider the impact that business strategies can have on a developer community, and vice versa. PAGE 05	<p>Open source is not a business model. It is a development and distribution model that is enabled by a licensing tactic. Vendors that build revenue streams around open source software for the most part do not choose between open source and proprietary development and licensing; they choose business strategies that attempt to make the best use of both open source and proprietary development and licensing models in order to maximize their opportunities for generating revenue and profit.</p>

OCTOBER, 2008

the 451 group Tier1Research

8000 THE 451 GROUP COMMERCIAL ADOPTION OF OPEN SOURCE

Open source is not a business model

“My view was always that open source is not a business model, nor an industry. Open source is just a production model and a distribution model. So any monetization must be added from the outside, i.e. there is no intrinsic business model for open source.”

Marten Mickos



Open source is not a business model

“There is no open source business model.”

Jim Whitehurst

*“I debated again yesterday with a
colleague on open source business models.
I don't believe there is such a thing.”*

Stephen Walli



Elements of an open source business strategy

Revenue triggers

- Commercial license
- Subscription
- Support services
- Other products/services
- Software services
- Custom development
- Advertising



End user licensing

- Dual licensing
- Open core
- Open complement
- Single open source
- Assembled open source
- Closed

Software license

- Reciprocal
- Permissive

Development model

- Public
- Private
- Public/private
- Vendor
- Community
- Aggregate



Defining community relationships

Copyright control

Vendor
Foundation
Distributed
Withheld

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Defining customer relationships

Revenue triggers

- Commercial license
- Subscription
- Support services
- Other products/services
- Software services
- Custom development
- Advertising



Copyright control

- Vendor
- Foundation
- Distributed
- Withheld

End user licensing

- Dual licensing
- Open core
- Open complement
- Single open source
- Assembled open source
- Closed

Elements of an open source business strategy

Revenue triggers

- Commercial license
- Subscription
- Support services
- Other products/services
- Software services
- Custom development
- Advertising



End user licensing

- Dual licensing
- Open core
- Open complement
- Single open source
- Assembled open source
- Closed

Software license

- Reciprocal
- Permissive

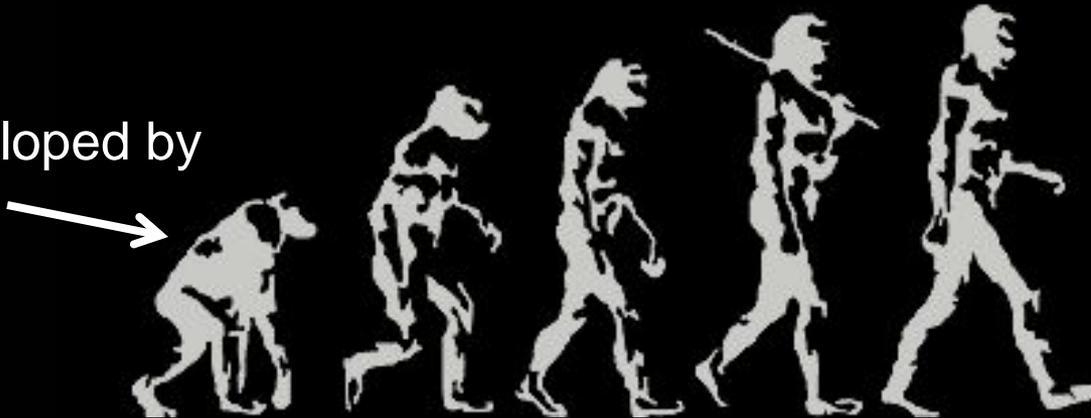
Development model

- Public
- Private
- Public/private
- Vendor
- Community
- Aggregate



The evolution of open source business strategies

Stage 1.0:
Software developed by
communities
of individuals.



the 451 group

Free or free?

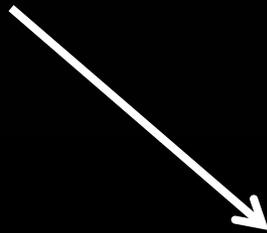
“There is no contradiction between selling copies and free software. In fact, the freedom to sell copies is crucial... Selling them is an important way to raise funds for free software development”

Richard Stallman

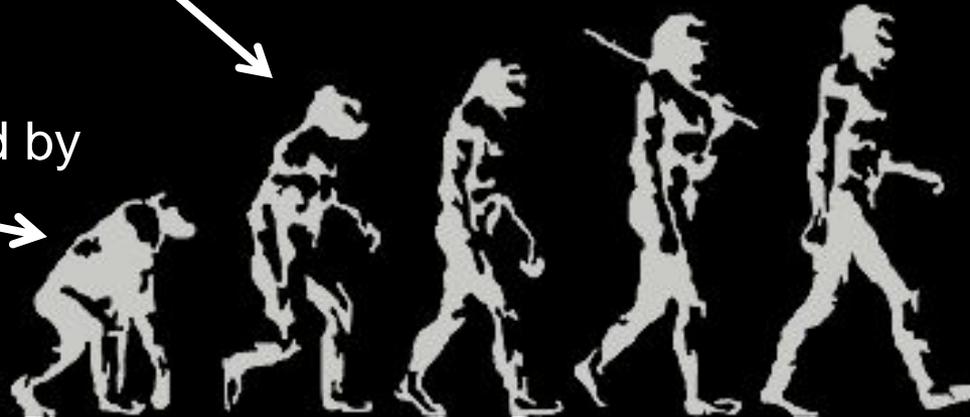


The evolution of open source business strategies

Stage 2.0: Vendors begin to engage with the existing developer communities.



Stage 1.0: Software developed by communities of individuals.



the 451 group

Products or services?

“The move to free eradicates the traditional concept of product, and shifts the value that a company can provide to customers to services.”

Michael Tiemann



Cygnus Solutions

Revenue triggers

Commercial license

Subscription

Support services

Other products/services

Software services

Custom development

Advertising

Copyright control

Vendor

Foundation

Distributed

Withheld

End user licensing

Dual licensing

Open core

Open complement

Single open source

Assembled open source

Closed

Software license

Reciprocal

Permissive

Development model

Public

Private

Public/private

Vendor

Community

Aggregate



Cygnus Solutions

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Linux vendors – generic

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Linux vendors – Caldera

Revenue triggers

Commercial license

Subscription

Support services

Other products/services

Software services

Custom development

Advertising

Copyright control

Vendor

Foundation

Distributed

Withheld

End user licensing

Dual licensing

Open core

Open complement

Single open source

Assembled open source

Closed

Software license

Reciprocal

Permissive

Development model

Public

Private

Public/private

Vendor

Community

Aggregate



Linux vendors – VA Research

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Linux vendors – Red Hat

Revenue triggers

Commercial license

Subscription

Support services

Other products/services

Software services

Custom development

Advertising

Copyright control

Vendor

Foundation

Distributed

Withheld

End user licensing

Dual licensing

Open core

Open complement

Single open source

Assembled open source

Closed

Software license

Reciprocal

Permissive

Development model

Public

Private

Public/private

Vendor

Community

Aggregate

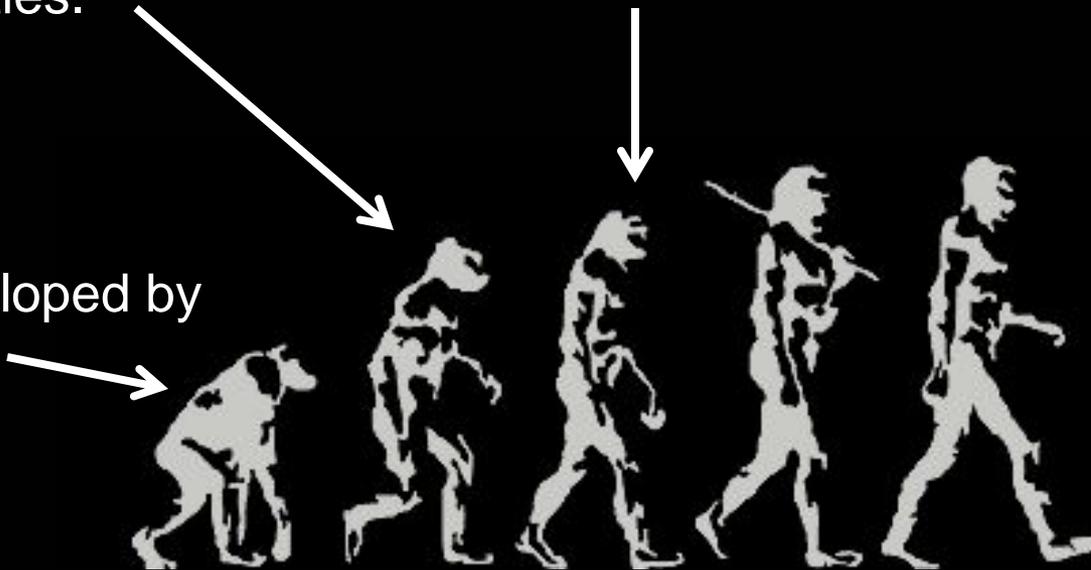


The evolution of open source business strategies

Stage 2.0: Vendors begin to engage with the existing developer communities.

Stage 3.0: Vendor-dominated open source development/distribution projects.

Stage 1.0: Software developed by communities of individuals.



the 451 group

Products or services?

“The move to services is not being led by open source companies themselves - they have modeled themselves on the companies they are trying to dislodge.”

Mark Shuttleworth



Dual licensing

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



MySQL

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Monty Program - current

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Monty Program - desired

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Monty Program - desired

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



OpenNMS - previously

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising

Copyright control

Vendor
Foundation
Distributed
Withheld

End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



OpenNMS - potentially

Revenue triggers

- Commercial license
- Subscription
- Support services
- Other products/services
- Software services
- Custom development
- Advertising



End user licensing

- Dual licensing
- Open core
- Open complement
- Single open source
- Assembled open source
- Closed

Software license

- Reciprocal
- Permissive

Development model

- Public
- Private
- Public/private
- Vendor
- Community
- Aggregate



OpenNMS - in actuality

Revenue triggers

Commercial license

Subscription

Support services

Other products/services

Software services

Custom development

Advertising

Copyright control

Vendor

Foundation

Distributed

Withheld

End user licensing

Dual licensing

Open-core

Open-complement

Single open-source

Assembled open-source

Closed

Software license

Reciprocal

Permissive

Development model

Public

Private

Public/private

Vendor

Community

Aggregate



Open core

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Open core (community controlled)

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

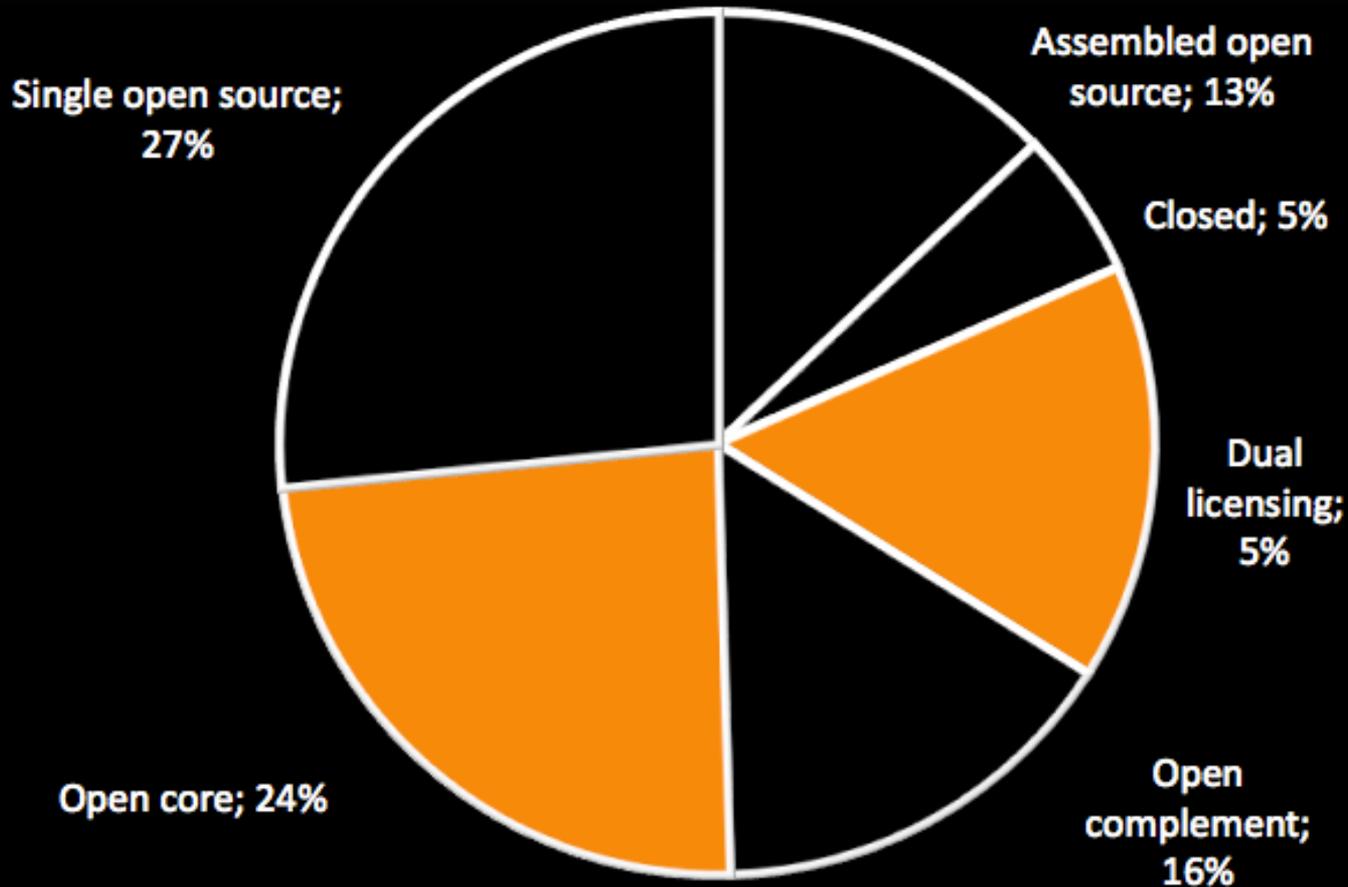
Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate

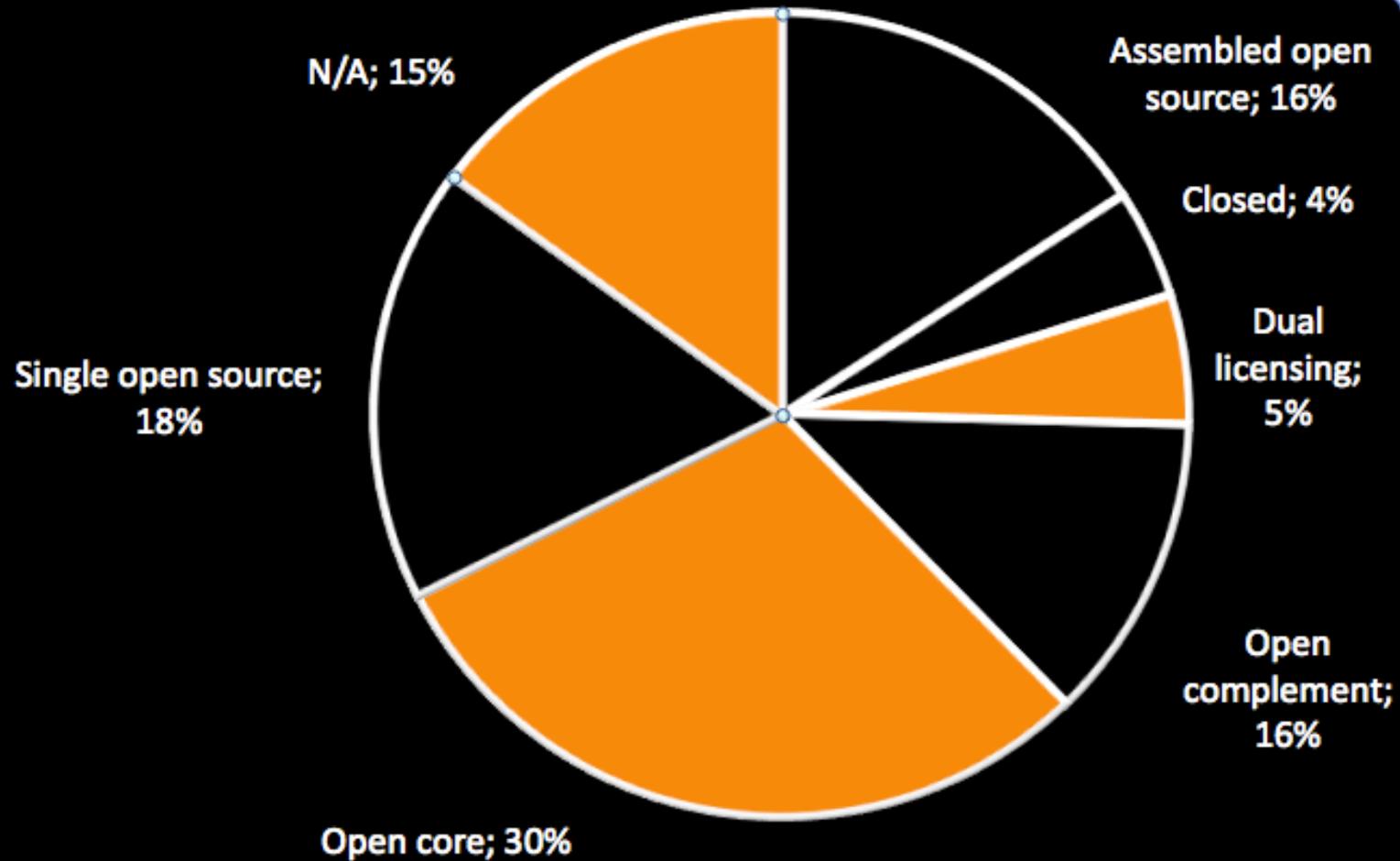
End user licensing strategies 2008



the **451** group

Sample: 114 "open source vendors"

End user licensing strategies 2010



the 451 group

Sample: 114 "open source vendors"

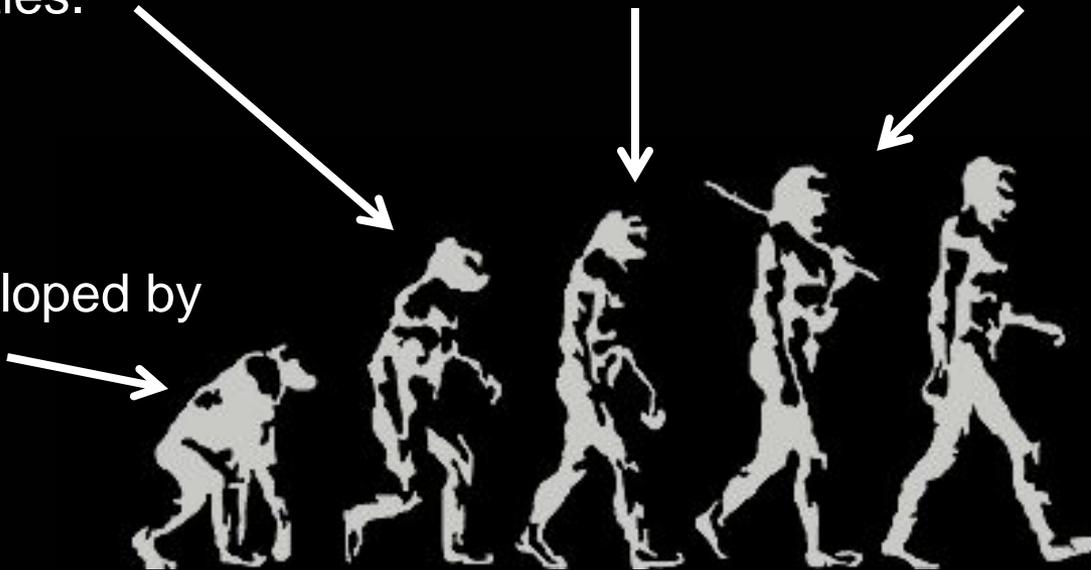
The evolution of open source business strategies

Stage 2.0: Vendors begin to engage with the existing developer communities.

Stage 3.0: Vendor-dominated open source development/distribution projects.

Stage 4.0 Vendor-dominated development communities.

Stage 1.0: Software developed by communities of individuals.



the 451 group

Proprietary or open source?

“I’m coming around to the idea that everything will be a blend of open source and proprietary software or services.”

Matt Asay



Open complement

Revenue triggers

Commercial license
Subscription
Support services

Other products/services

Software services
Custom development
Advertising

Software license

Reciprocal
Permissive



End user licensing

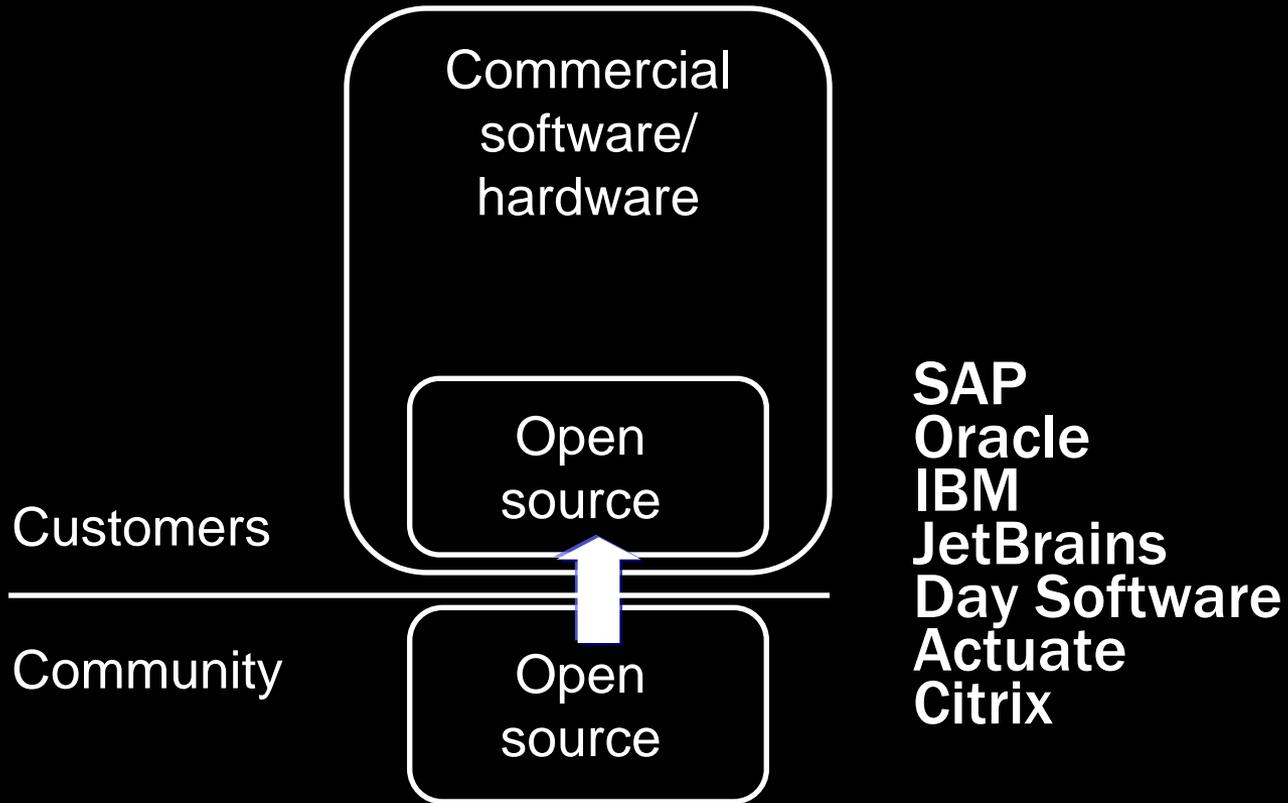
Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Development model

Public
Private
Public/private
Vendor
Community
Aggregate

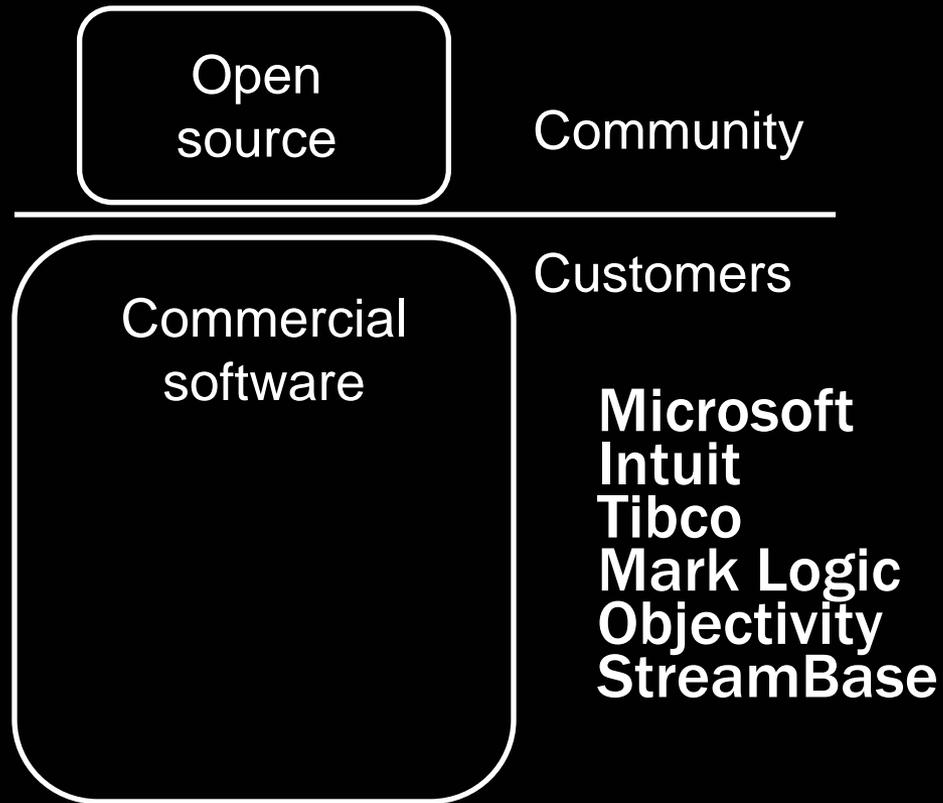


Open inside



the 451 group

Open edge



The evolution of open source business strategies

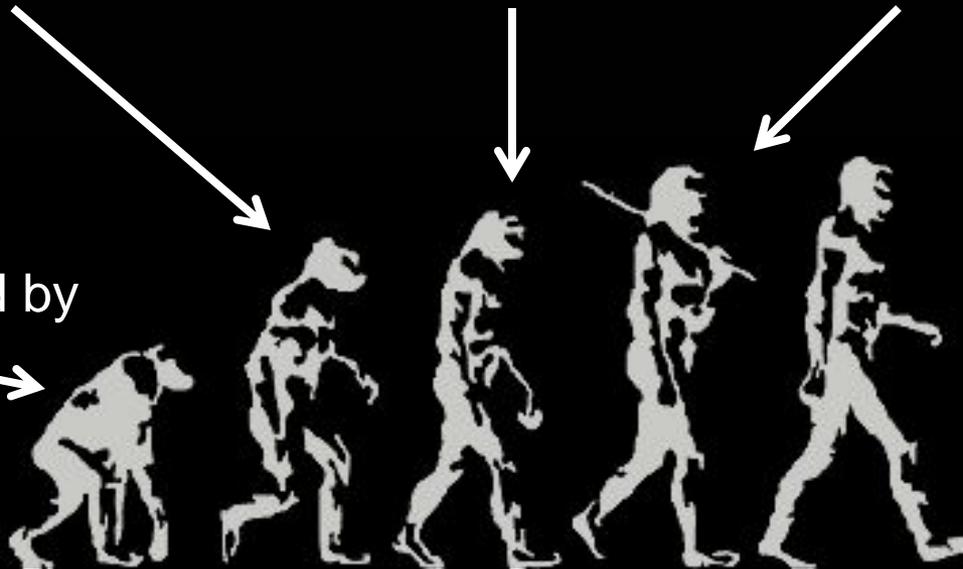
Stage 2.0: Vendors begin to engage with the existing developer communities.

Stage 3.0: Vendor-dominated open source development/distribution projects.

Stage 4.0 Vendor-dominated development communities.

Stage 5.0
The future of commercial open source strategies.

Stage 1.0:
Software developed by communities of individuals.



the 451 group

Software or services?

“The industry has taken open source to one logical conclusion and has crowned ‘services’ as the only important software outcome... The cloud takes open source to its logical conclusion, crowning services as king, rather than fettering us to a discussion of software.”

Matt Asay



Google/Amazon/Twitter/Facebook

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Open source license trends

GPLv2 % of projects 07/09
50.6%

GPLv2 % of projects 03/10
48.9%



Source: Black Duck Software

Open source license trends

GPLv2 % of projects 07/09
50.6%

GPLv2 % of projects 03/10
48.9%

GPLv2 projects 07/09
93,730

GPLv2 projects 03/10
97,042



Open source license trends

GPLv2 % of projects 07/09
50.6%

GPLv2 % of projects 03/10
48.9%

GPLv2 projects 07/09
93,730

GPLv2 projects 03/10
97,042

AGPLv3 projects 07/09
199

AGPLv3 03/10
231

the **451** group

Source: Black Duck Software

Alfresco

Revenue triggers

- Commercial license
- Subscription
- Support services
- Other products/services
- Software services
- Custom development
- Advertising



End user licensing

- Dual licensing
- Open core
- Open complement
- Single open source
- Assembled open source
- Closed

Software license

- Reciprocal
- Permissive

Development model

- Public
- Private
- Public/private
- Vendor
- Community
- Aggregate



Coupa

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



Citrix GoToManage (Paglo)

Revenue triggers

Commercial license
Subscription
Support services
Other products/services
Software services
Custom development
Advertising



End user licensing

Dual licensing
Open core
Open complement
Single open source
Assembled open source
Closed

Software license

Reciprocal
Permissive

Development model

Public
Private
Public/private
Vendor
Community
Aggregate



From support services to software services

“Established vendors have adopted open source and cloud computing has changed the rules about how businesses adopt technology. The long-term opportunities are not in open source products and services, but products and services that are built on open source.”

The 451 Group



The future of open source business strategies

Update on open source
-related business strategies

Survey of 1,200 CAOS
user community members

Scheduled:
October 2010

Comments/suggestions:
twitter.com/maslett
matt.aslett@the451group.com

the 451 group
Analyzing the Business
of Enterprise IT Innovation

OPEN SOURCE IS **still** NOT A BUSINESS MODEL

How Vendors Generate Revenue from Open Source Software

Open source software is freely available, but vendor lock-in comes at a price. This report assesses the development, licensing and revenue-generation strategies used by vendors to make money from open source software.

CAOS COMMERCIAL ADOPTION OF OPEN SOURCE

4 FINDINGS	5 IMPLICATIONS	1 BOTTOM LINE
<ul style="list-style-type: none">The majority of vendors analyzed utilize some form of commercial licensing to distribute, or generate revenue from, open source software. PAGE 10Half of the vendors assessed in this report are combining code developed via open source projects with software developed out-of-sight of open source project members. PAGE 11Vendors using hybrid development and licensing models are balancing higher development and marketing costs with the ability to increase revenue-generation opportunities from commercially licensed software. PAGE 20The license used for an open source project (proprietary or permissive) has a strong influence on development, vendor licensing and revenue-generation strategies. PAGE 20	<ul style="list-style-type: none">The line between closed and open source has blurred as FOSS is embedded in proprietary products and commercial extensions have been added to FOSS. PAGE 27Vendors must choose complementary development, licensing and revenue strategies in order to maximize revenue-generation opportunities. PAGE 28Customers make sure that they are aware of vendors' strategies so they can understand and predict the behavior of vendors (encouraging them to become paying customers). PAGE 27Increased use of commercial licensing could create tension between the expectations of potential customers and commercial reality. PAGE 46Vendors need to consider the impact that business strategies can have on a developer community, and vice versa. PAGE 50	<p>Open source is not a business model. It is a development and distribution model that is enabled by a licensing tactic. Vendors that build revenue streams around open source software for the most part do not choose between open source and proprietary development and licensing; they choose business strategies that attempt to make the best use of both open source and proprietary development and licensing models in order to maximize their opportunities for generating revenue and profit.</p>

OCTOBER, 2008

the 451 group Tier1Research

888 THE 451 GROUP COMMERCIAL ADOPTION OF OPEN SOURCE