

Innovation through Optimal Licensing

Plone Symposium

March 10, 2006

Geoffrey Parker
Tulane University

Marshall Van Alstyne
Boston University & MIT

Sponsored by NSF IIS-0338662

Agenda

<i>Topic</i>	<i>Time</i>
■ Motivation & Examples	10
■ Redefining network effects <ul style="list-style-type: none">– Non-mathematical intuition & biz logic	20
■ Stimulating innovation	20

US Constitution – Article 1, Section 8, Clause 8 : “The Commerce Clause”

Congress shall have power to promote the progress of science and useful arts, by securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries.

The Debate

Longer is Better

- Infinitely long but narrow patents
 - Gilbert & Shapiro '90
- Infinitely renewable ©
 - Landes & Posner '03
- Even “perfect” control better for incentives
 - Wagner '03
- Capturing sequential innovation
 - Green & Scotchmer '95

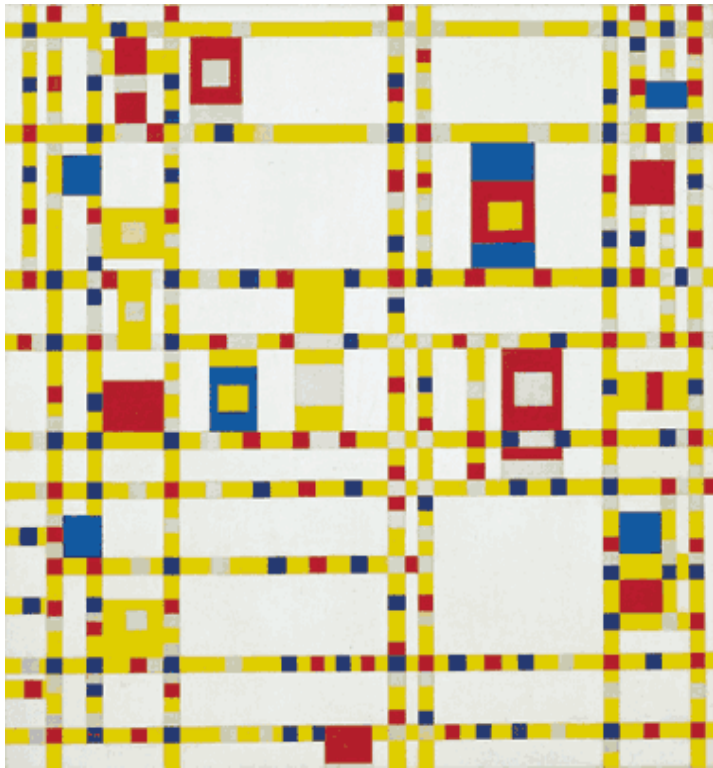
Free / Open is Better

- Fundamental right of access
 - Stallman '92
- Tragedy of the “AntiCommons”
 - Heller & Eisenberg '98
- Collective production
 - Benkler '02
 - Hippel & Krogh '03
- Open Science (enablement)
 - David '04

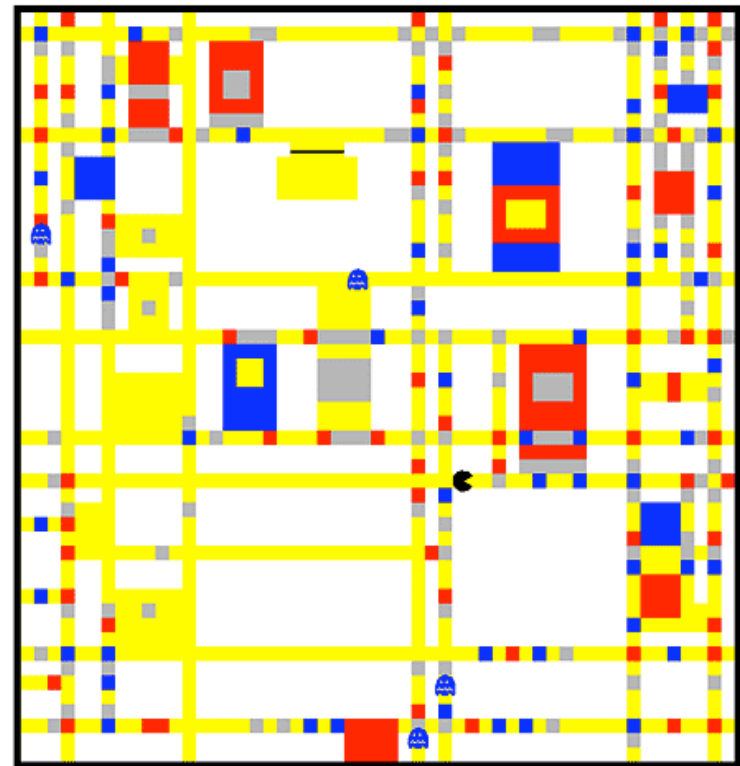
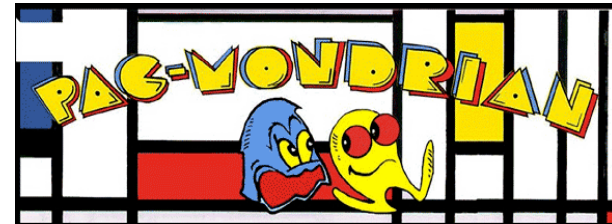
Motivating Issues

- OSS licenses destroy economic incentives to innovate.
 - Even Raymond admits that such licenses only permit *indirect* business models like sale of services.
- To build on a code base, a developer needs access *and* permission. Hold-up and monopsony curb innovation.
- Neither patent nor copyright law adequately solve the problems of full disclosure & enablement for software.

Reusable Art



Piet Mondrian. *Broadway Boogie Woogie*. 1942–43. Oil on canvas, 50 x 50"(127 x 127 cm)



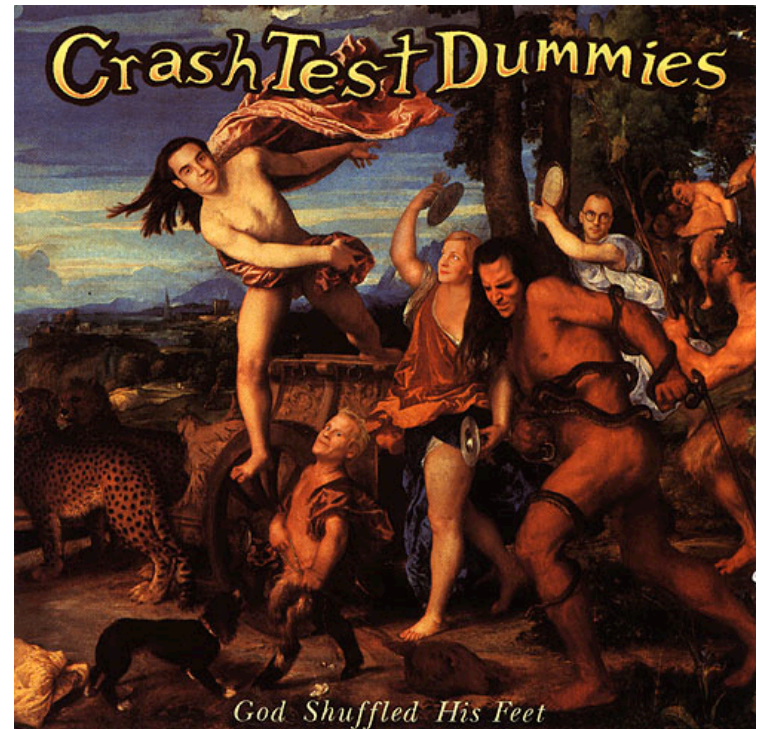
Sarah Boxer "Chomp if you like Art" 12-27-04 NYT

Reusable Art



Copyright © 2002 The National Gallery, London. All rights reserved.

Bacchus & Ariadne by Titian



Reusable HTML

WELCOME TO HICSS - Mozilla Firefox

http://www.hicss.hawaii.edu/



WELCOME TO HICSS

HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES

Sponsored by College of Business
University of Hawai'i at Mānoa

HICSS-38
January 3-6, 2005
Hilton Waikoloa Village
Big Island, Hawaii



[HICSS-38 Conference Information](#)
[Paper Presentation Schedule](#)

[On-Site Registration Available](#)

[Hilton Waikoloa Reservation Form](#)
[Waikoloa Beach Marriott](#)

[Previous Conferences \(HICSS 28-37\)](#)
[Best Papers \(HICSS-37\)](#)











HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES - Mozilla Firefox

http://www.hicss.hawaii.edu/prevconf.htm

Previous Conferences

(All previous conference materials are located on the Web server at the University of Hawaii; none are available on the mirror sites.)

[HICSS-28](#) [HICSS-29](#) [HICSS-30](#) [HICSS-31](#) [HICSS-32](#) [HICSS-33](#) [HICSS-34](#) [HICSS-35](#) [HICSS-36](#)
[HICSS-37](#)

Mash-Ups

- I'm awed by the power of the people: the hackers, the experimenters, the strangers who help strangers in the online cellphone forums. If the cellphone companies were smart, they'd realize that these customers are their allies, not their enemies.

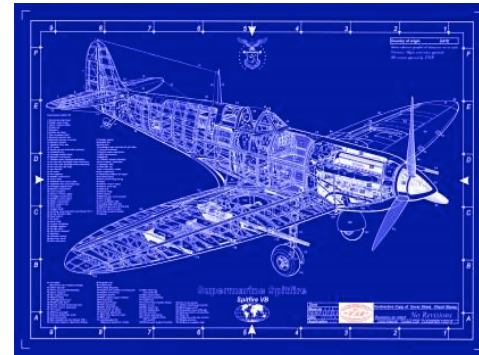
David Pogue NYT 10/27/05



Source: NYT 10/20/05 – “Journey to a Thousand Maps Begins With an Open Code”

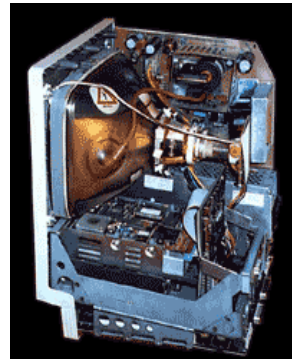
What do the aircraft, auto & computer industries have in common?

Autos



Aircraft

Computers



One Answer: As in aerospace, autos and computers are moving from go-it-alone systems toward integrated systems of best-of-breed parts.

Platform – Any architecture or standard with sufficient intrinsic value that the interests of buyers and suppliers commingle.

Computer hardware, operating systems, auction websites, HDTV, aircraft, streaming video, electronic health records, ...

Question: What are “network externalities?”

Hint: they matter for platforms...

Reinterpreting Network Externalities

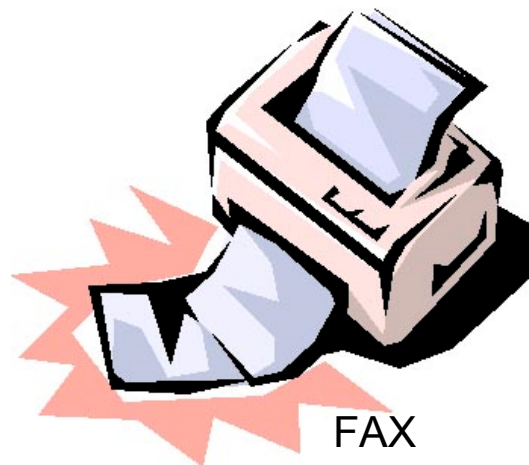
- “Network Externalities” are demand economies of scale.
- They imply at least some level of interaction as when I email you, or you FAX or IM me.



Phone



E-Mail



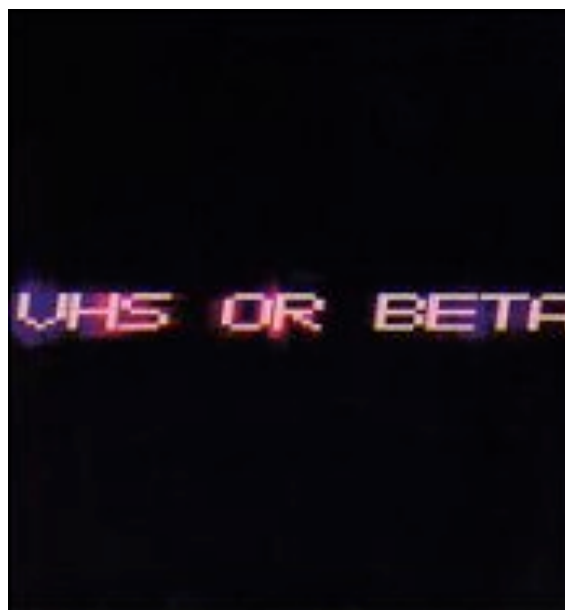
FAX



IM

Reinterpreting Network Externalities

- Where is the *interaction* when your neighbor rents “Lord of the Rings”?
- In fact, his rental may mean you have to wait!



**A “2-sided” network externality
crosses markets from consumers to
developers or developers to consumers**

Why do profit making firms give so much away?

- Adobe Acrobat Reader
- RedHat Linux
- Microsoft Internet Explorer
- Fidelity Retirement Planner
- Wolfram Mathematica Reader
- Bungie/ID Game Level Editors
- Intel Video Morphing Software
- Kodak Digital Camera Scripts
- Lexis / Nexis Law Student Access
- Sun Microsystems Star Office

2-Sided Networks Include

- credit cards
- streaming media
- night clubs
- computer operating systems
- shopping malls
- instant messaging services
- video games
- fuel-cell powered cars
- and much more...

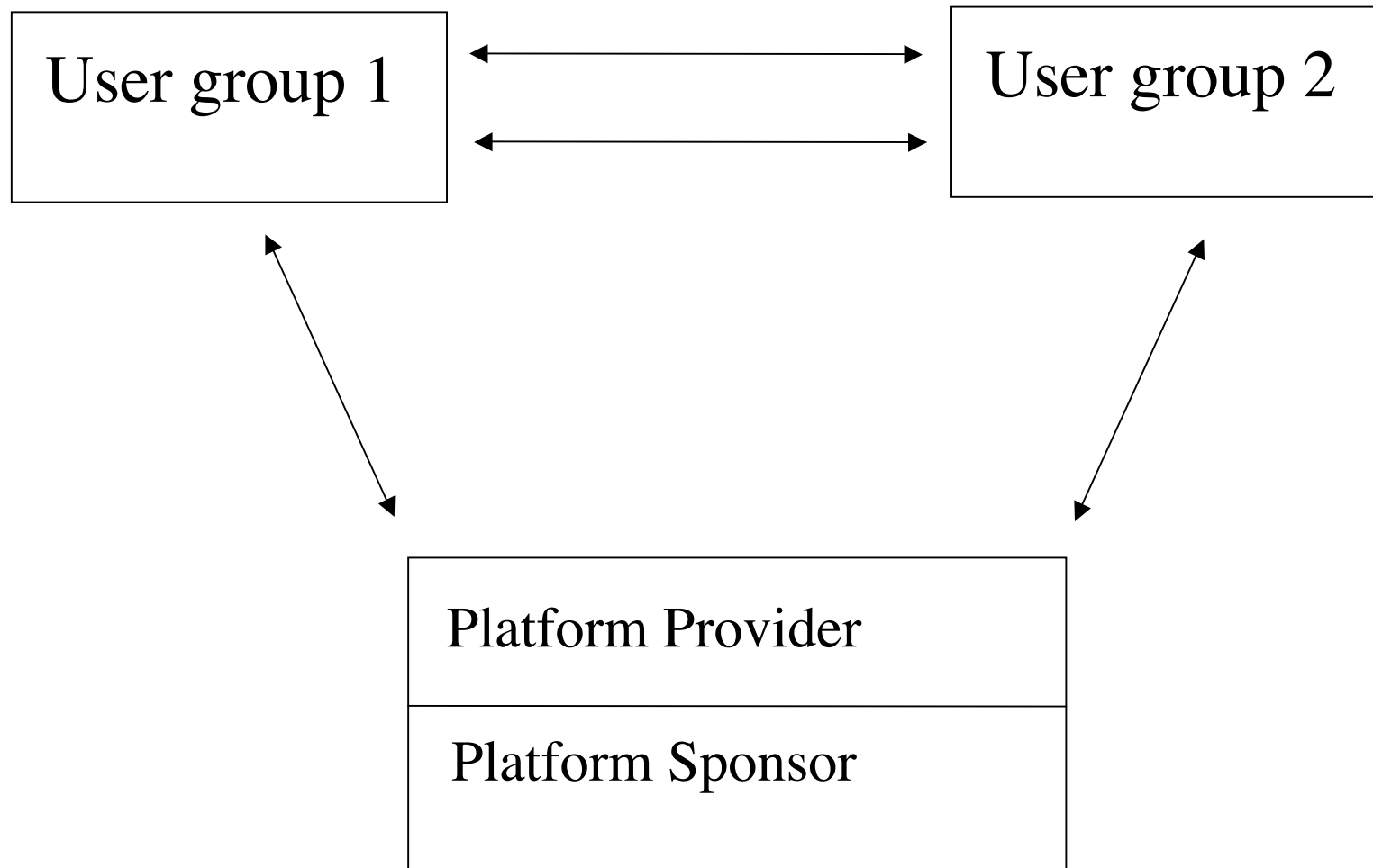
Readily identifiable platform markets

Product Category	Mkt 1 Product	Intermediary	Mkt 2 Product
Portable Documents	Document reader*	Adobe	Document Writer
Credit Cards	Consumer credit*	Issuing bank	Merchant Processing
Operating Systems	Complementary Applications	Microsoft, Apple, Sun	Systems Developer Toolkits*
Plug-Ins	Applications Software	Microsoft, Adobe	Systems Developer Toolkits*
Ladies' Nights	Men's Admission	Bars, Restaurants	Women's Admission*
TV Format	Color UHF, VHF, HDTV*	Sony, Phillips, RCA	Broadcast Equipment
Advertisements	Content*	Magazines, TV, Radio	Advertisers
Computer games	Game Engine/ Player	Games Publishers	Level Editors*
Auctions	Buyers*	E-bay, Christie's, Sotheby's	Sellers
Streaming Audio/Video	Content*	RealPlayer, Microsoft, Apple	Servers

* Indicates which market is discounted, free or subsidized.

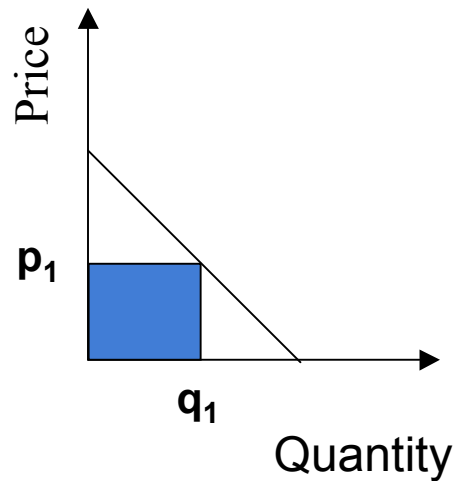
Source: Parker & Van Alstyne 2002

Distinct user group consumption affects value by interacting through a platform

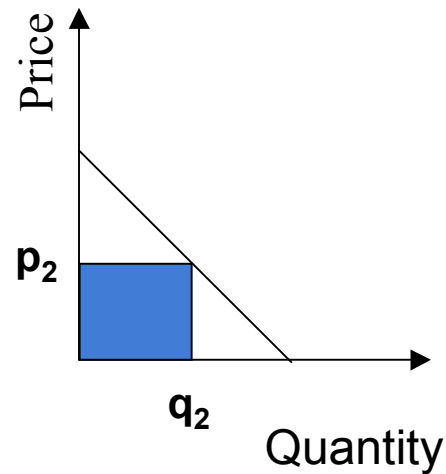


Consider profits in two markets

Market One
(Acrobat Reader)



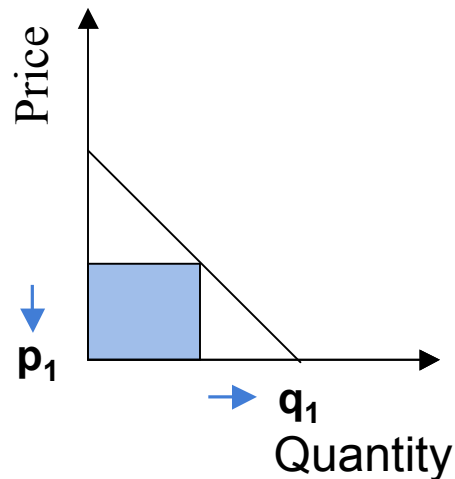
Market Two
(Acrobat Distiller)



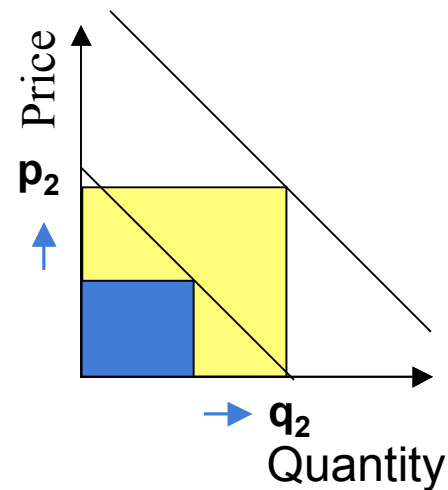
Initially, there are profits to be made in both markets.

Consider profits in two markets

Market One
(Acrobat Reader)



Market Two
(Acrobat Distiller)

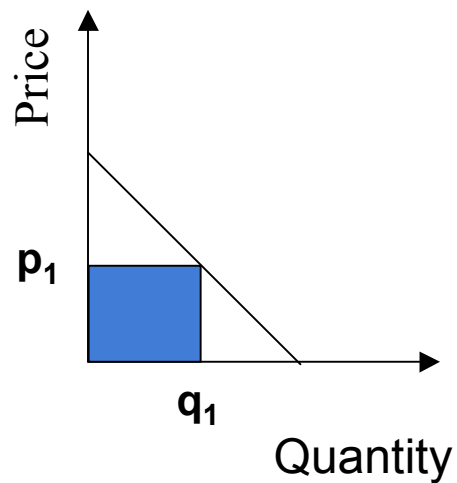


Initially, there are profits to be made in both markets.

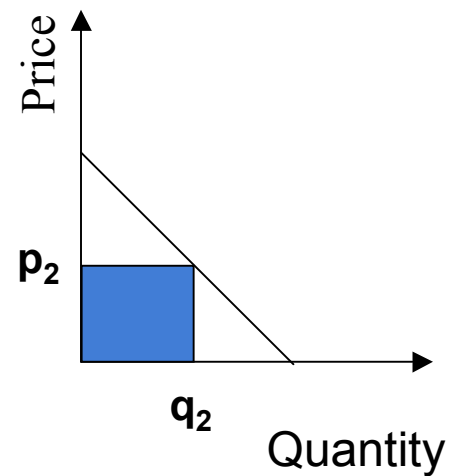
But subsidizing market one can increase demand and profits in market two more than the loss in market one.

Which market to subsidize

Consumer Market



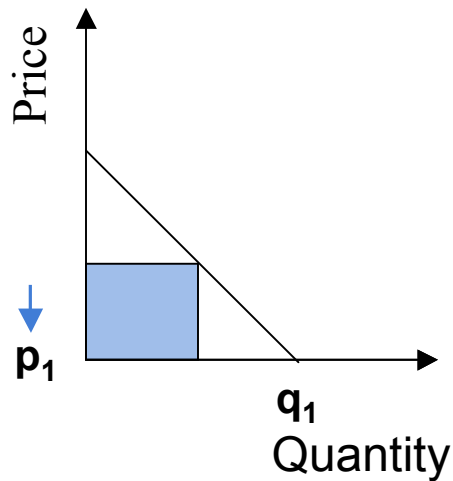
Developer Market



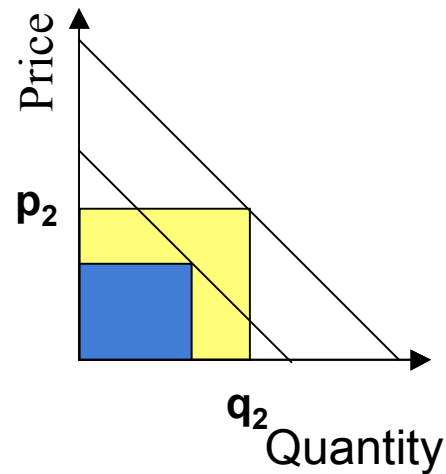
Consider which market creates more surplus.

Which market to subsidize

Consumer Market



Developer Market

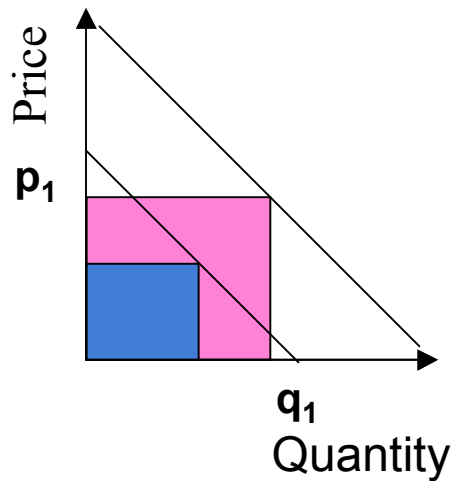


Consider which market creates more surplus.

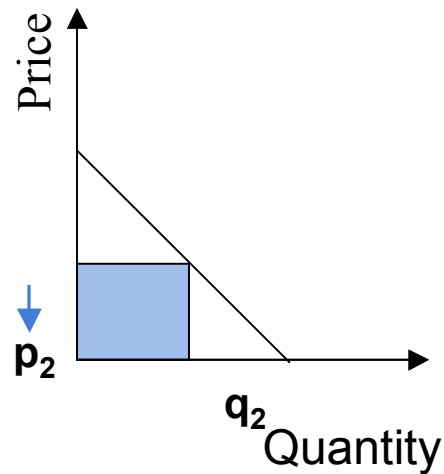
Subsidize the one that creates more surplus in the cross market.

Which market to subsidize

Consumer Market



Developer Market

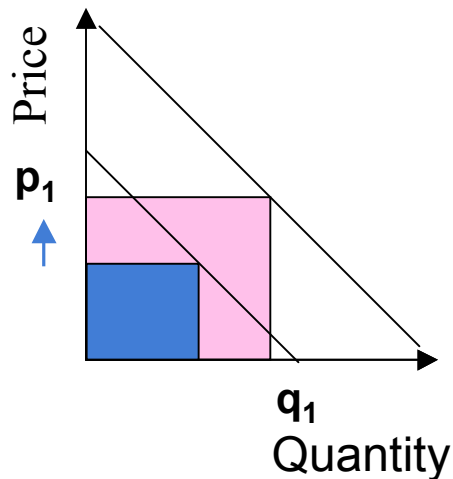


Consider which market creates more surplus.

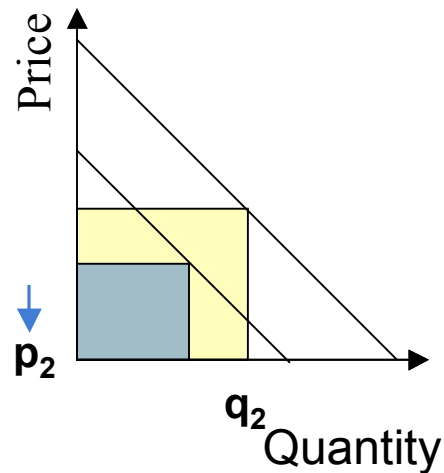
Subsidize the one that creates more surplus in the cross market.

Which market to subsidize

Consumer Market



Developer Market

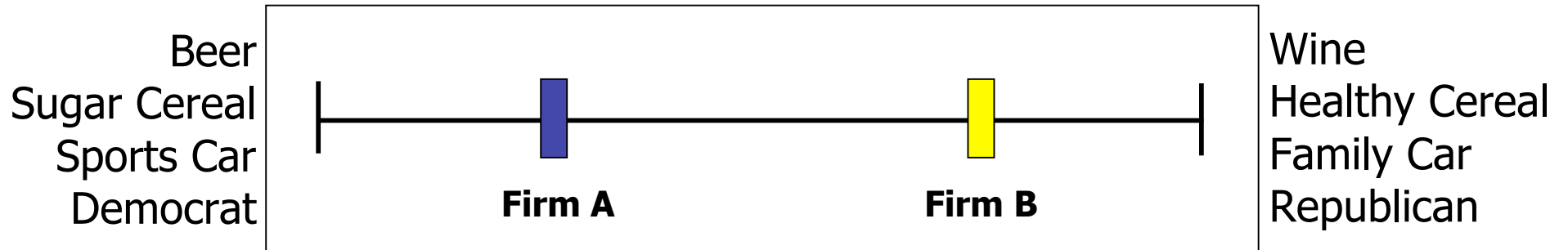


Consider which market creates more surplus.

Subsidize the one that creates more surplus in the cross market. Here $>$ so subsidize developers.

**Now introduce product
competition.**

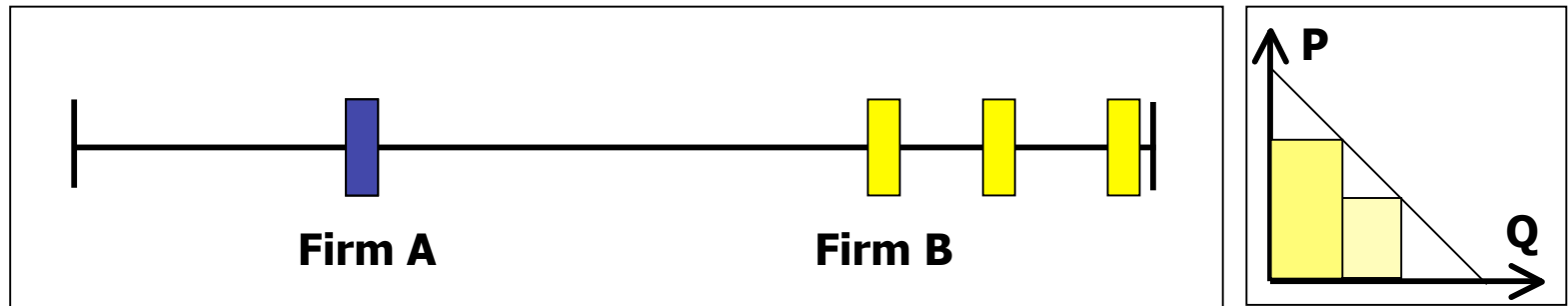
Economics of Product Competition



$$\Pi = P * Q$$

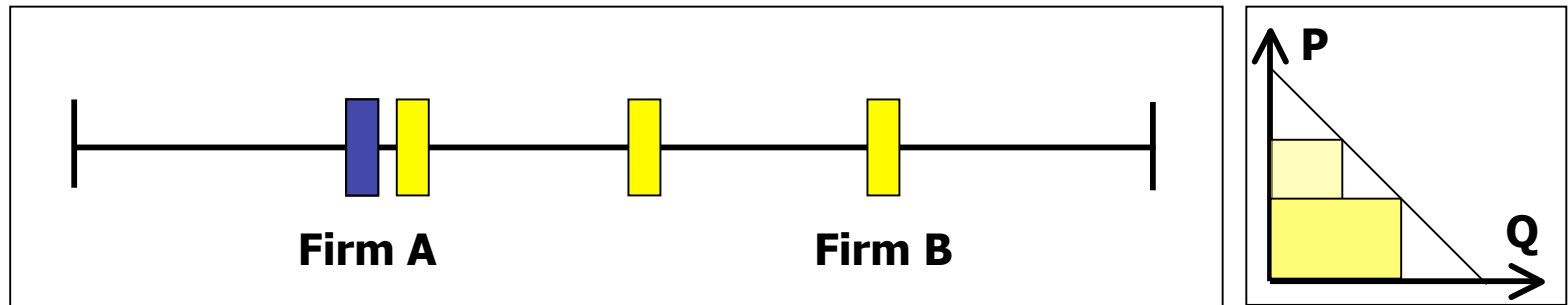
Firms can max Π by either
building (1) market power \Rightarrow
high P (2) market share \Rightarrow
high Q

To Gain Market Power



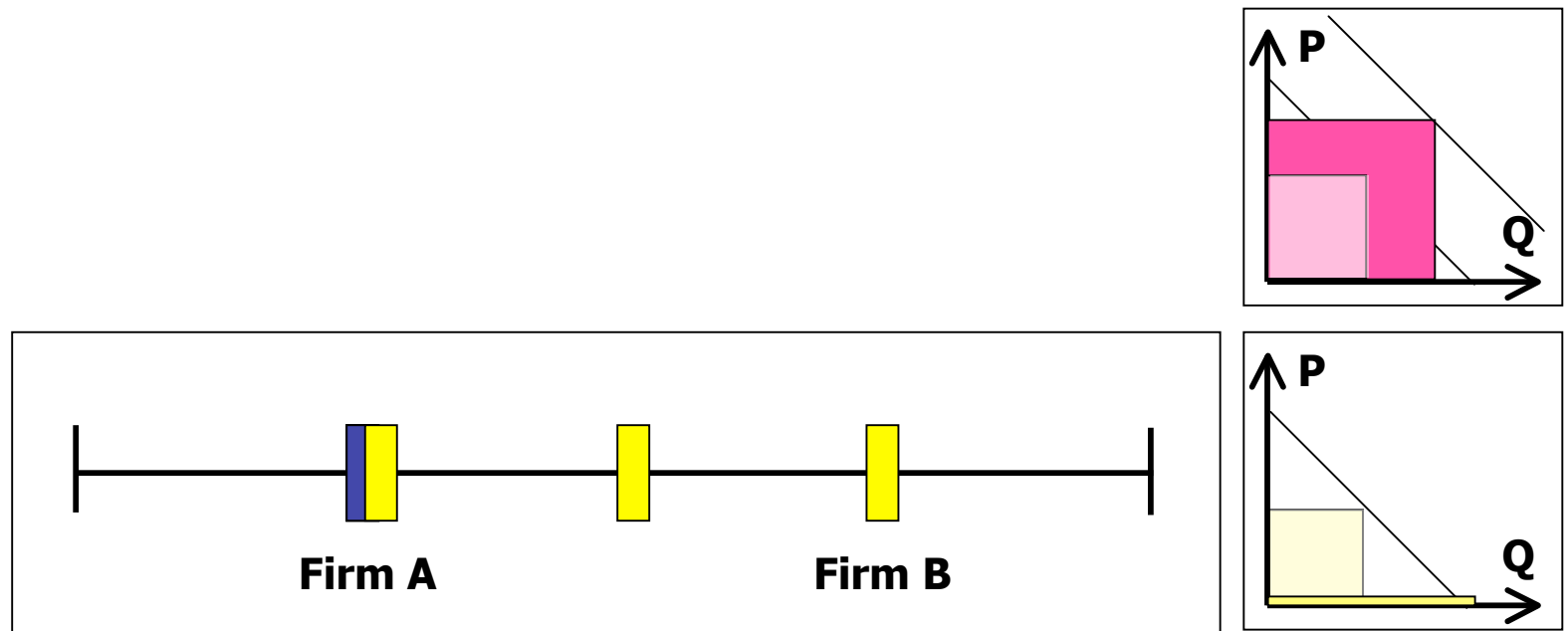
Differentiate your product
from the competitor

To Gain Market Share



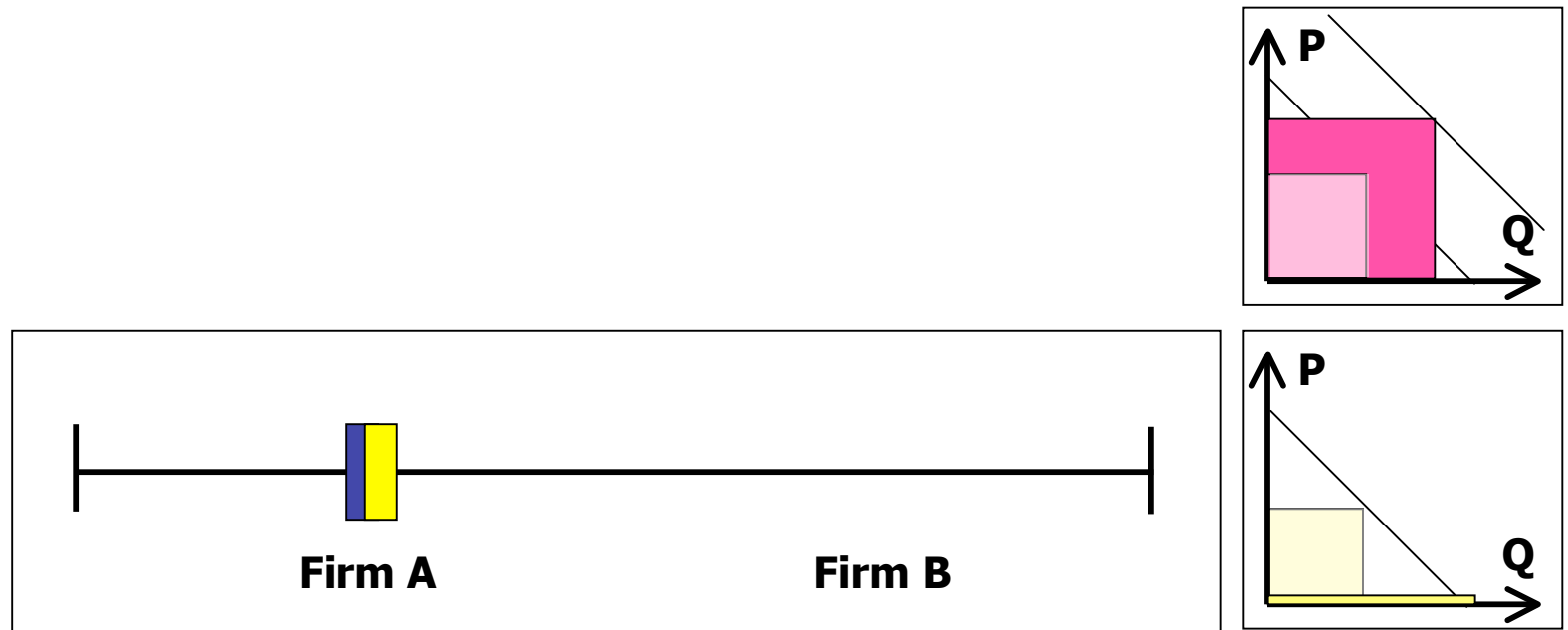
Position between competitor
and largest block of
consumers

Competitive Complement



Product complementarity justifies seeking market share *because* a price of zero increases profits in the coupled market.

Competitive Complement



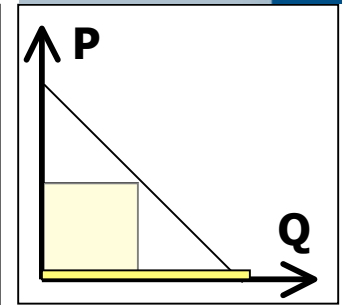
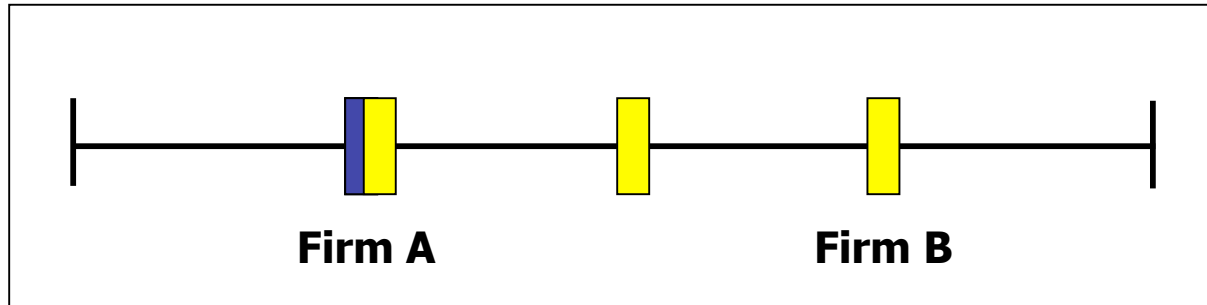
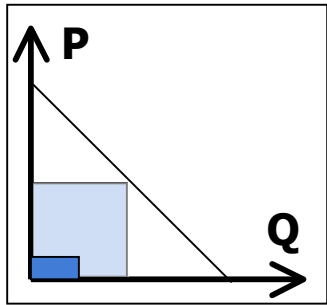
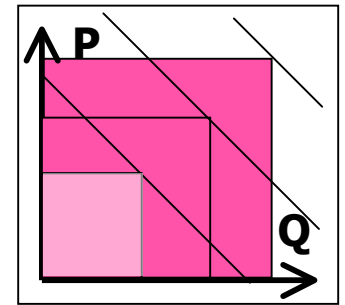
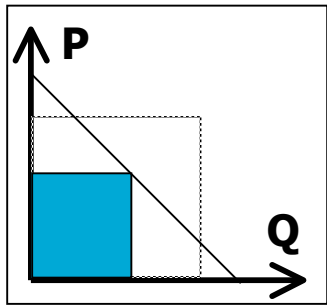
Examples:

MS Office
Java
MediaPlayer

Star Office =>
Internet Explorer =>
QuickTime/iTunes =>

Sun OS
Microsoft OS
iPods

Competitive Substitute



Product substitutability justifies seeking market share *because* a price of zero decreases competitive interference on another product.

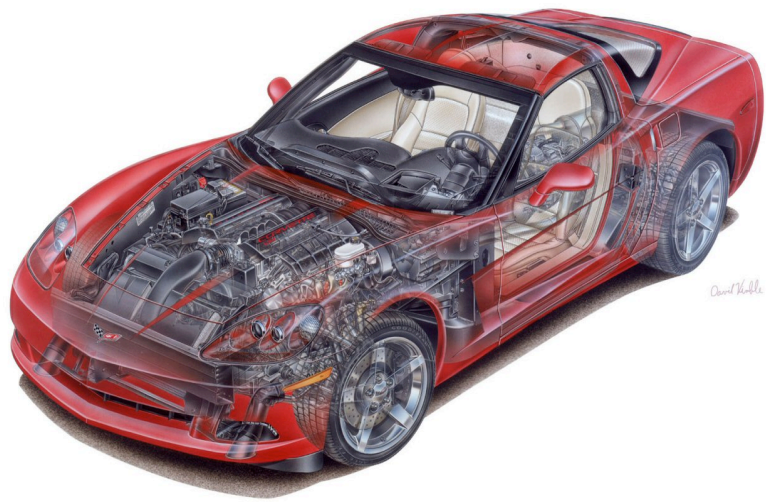
Strategic Outcome

- The product design results are almost identical. Firm *B* chooses $P_1 \leq 0$ - the market with competition- in order to *create a free goods barrier to entry*.
- The reason, however, is different. For complementary goods, Firm *B* sells more. For substitute goods, Firm *B* stems losses.

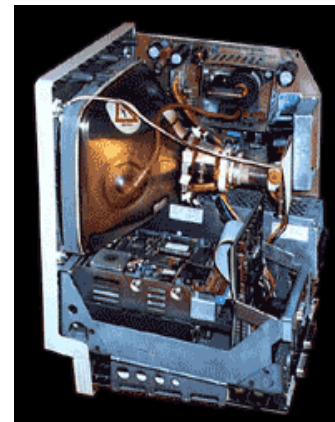
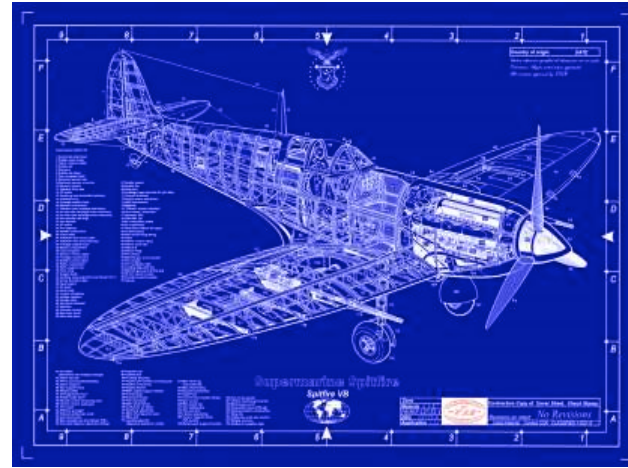
“Microsoft would not have given IE away ..., nor would it have taken on the high cost of enlisting firms in its campaign to maximize IE’s usage share and limit Navigator’s, had it not been focused on protecting the applications barrier [to its operating system]”

*Judge Thomas Penfield Jackson
Findings of Fact, 11-5-99*

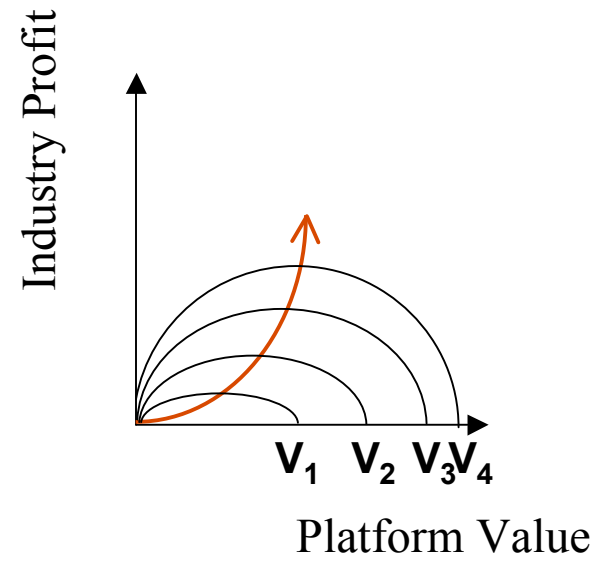
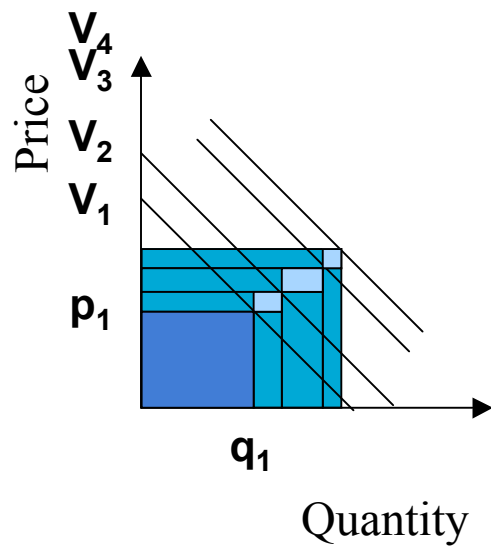
Back to platforms



David White



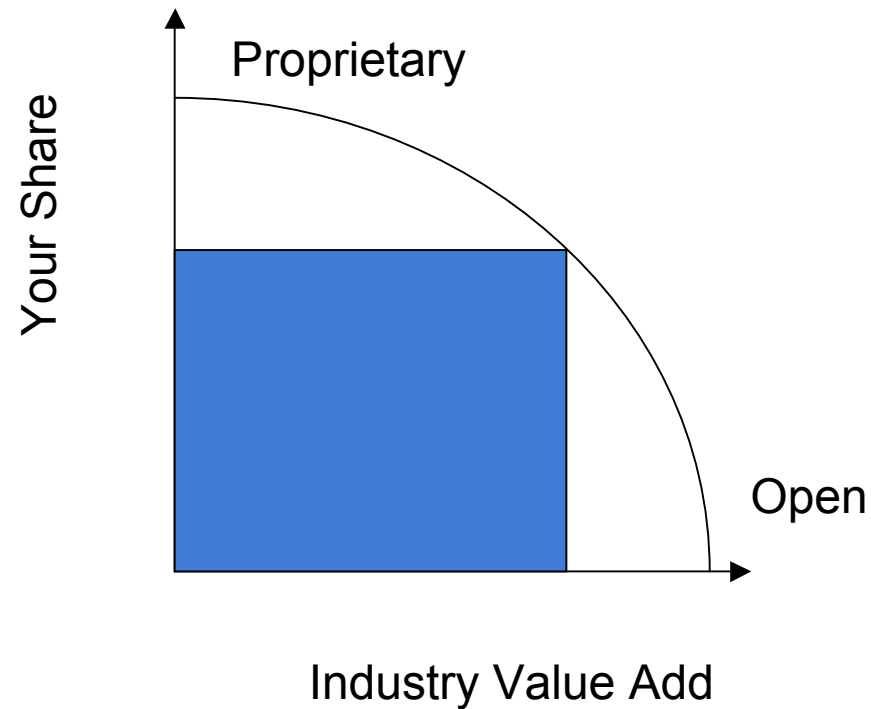
Creating profits on your platform



Question: So *how* do you stimulate innovation on your platform?

One answer: manage your platform standards, specs & IP to promote it...

Openness vs. Control



Your reward = (Value added to industry) x (Your share)

Source: Shapiro & Varian '99


Information Asymmetry – Any situation in which one party is better informed than another. Someone knows something you don't (or vice versa)

Version 2 : Who are the best developers and what are their ideas?

Issue

- To build on a platform, a developer needs access *and* permission. Hold-up and monopsony curb innovation.
- Open source style licenses grant default access and permission but destroy economic incentives to innovate.

Consider a Flexible License

- A default contract offered to anyone.
- Original author opens portions platform to public, keeping a key complement proprietary.
- Offers open source style license but forces disclosure on derivative works after T years.
- Inventor profits from
 -  sales of complementary good
 - New versions: $5 \times 10^9 > 95 \times 10^6$

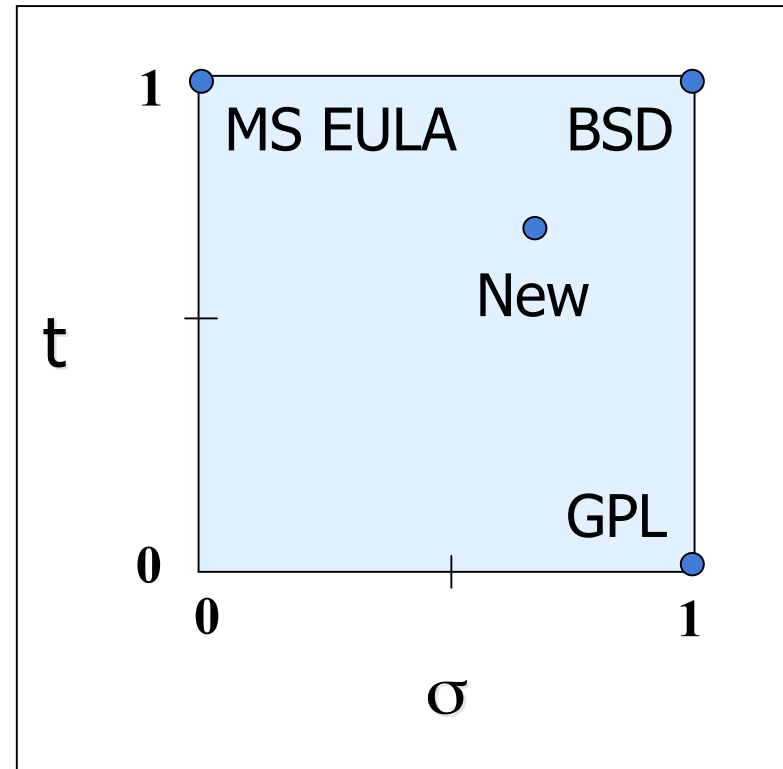
OSS Classification

Type	Free	Redist	Unlimited Use	Src. Avail.	Src. Mod.	Public CheckIn	Viral
Commercial							
Trial SW	X						
Shareware	X	X					
Royalty Free Bin.	X	X	X				
Royalty Free Lib.	X	X	X	X			
Open Src (BSD)	X	X	X	X	X		
Open Src (Apache)	X	X	X	X	X	X	
Open Src (Linux/GNU)	X	X	X	X	X	X	X

Source: Vinod Valloppillil / The Halloween Document

Most licenses are corner solutions

- Let $\sigma \in [0, 1]$ be the measure of openness.
- Let $t \in [0, 1]$ be the time to release derivative works.



Key Tradeoffs

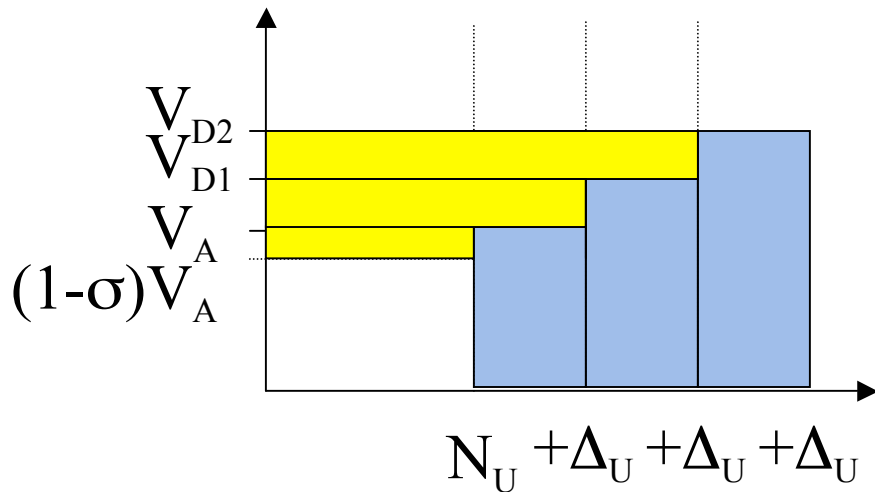
■ Increasing σ

- Diminishes profits
- Promotes adoption

■ Increasing t

- Delays adoption & retards access
- Promotes innovation

Coding on the shoulders of giants...



Platform authors can charge for V_A .

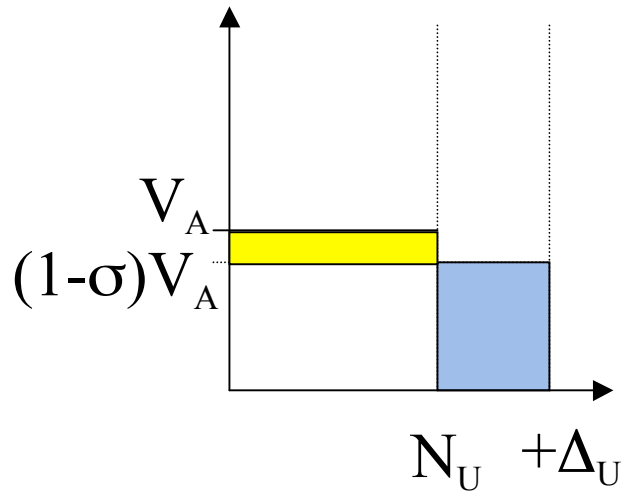
But can't charge for fraction σ given away.

Opening the code promotes adoption and opportunities for adaptation.

Developers add value.

Opening the code promotes adoption and opportunities for adaptation.

Repeat...



How can we model the feedback from
Developers \rightarrow Consumers \rightarrow Developers ...?

Tatonnement:

$$\sigma V_a \rightarrow e_{DU} \sigma V_a \rightarrow e_{UD} e_{DU} \sigma V_a \rightarrow e_{DU} e_{UD} e_{DU} \sigma V_a \rightarrow \text{etc.}$$

Defines Cauchy sequences: $\Delta_D = \frac{\sigma V_a}{1 - e_{DU} e_{UD}}$ $\Delta_U = \frac{e_{DU} \sigma V_a}{1 - e_{DU} e_{UD}}$

This provides flexible business models

- Direct consumer sales
 - Office Suites (word processing, spreadsheet, presentation)
 - Graphics packages (Photoshop, Illustrator, ...)
 - DB & Statistical tools (Access, Oracle, Stata, ...)
- Indirect growth in sales
 - Operating System APIs
 - Office Suite plug-ins
 - 3rd party DB & Stat add-on packages
- Indirect developer royalties
 - Game cartridges
 - Streaming media
 - Flash & PDF distiller

When Openness “Wins”

Innovation Test:

$$M_u V_a (1 + N_d \Delta) > N_u \quad \text{where} \quad \Delta \equiv k(1 - \delta)\delta$$

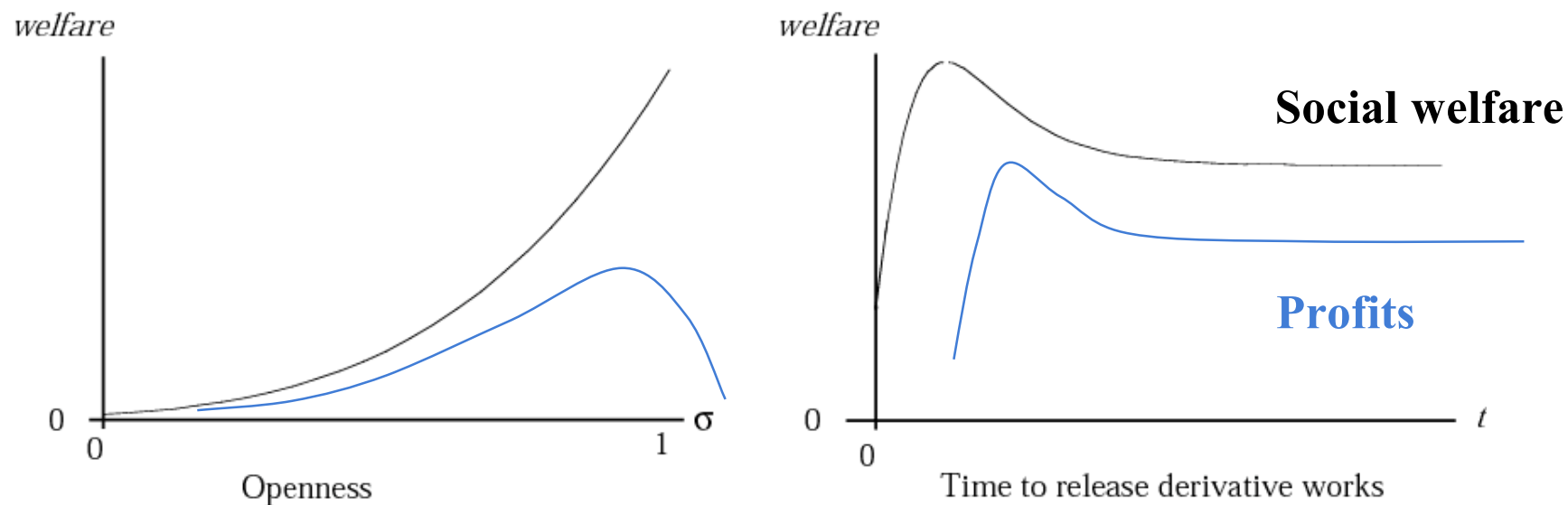
For any given user base N_u , there exist user side network effects M_u , platform values V_a , or developer value added, $N_d \Delta$ that justify opening the code $\sigma > 0$.

Optimum absent Innovation:

$$\sigma^+ = \frac{1}{2} - \frac{N_u}{M_u V_a}$$

Even if there is no innovation with reuse, s.t. $k = 0$, user side network effects are sufficient to justify opening the code, $s > 0$.

Profits & Social Welfare Can Rise



- Social welfare **and** profits **both show clear optimal values.**
- t can dynamically adjust to industry clockspeed.
- Implies a hybrid can be better for profits *and* welfare.

So Why Don't Developers “Do The Right Thing” Naturally?

The answer is a prisoner's dilemma.

arising endogenously from:

- 🔗 Larger reusable code base complements development.
- 🔒 Private desire to charge lengthens t .

		<i>Developer B</i>	
		Hoard	Free
<i>Developer A</i>	Hoard	(π^{HH}, π^{HH})	(π^{HF}, π^{FH})
	Free	(π^{FH}, π^{HF})	(π^{FF}, π^{FF})

$\pi^{HH} > \pi^{FH}$ $\pi^{HF} > \pi^{FF}$

This implies that contracts must enforce disclosure and some governance structure is better than anarchy

Issue 1

Problem: OSS licenses destroy economic incentives to innovate.

Proposal: Return sufficient pricing power to developers to restore incentives. At a minimum, benefits to developers should cover direct expenses and opportunity costs.

Issue 2

Problem: To build on a base of code, a developer needs not only access, but also permission. Hold-up and the monopsony problem reduce innovation incentives.

Proposal: As with GPL, offer a *default* contract that allows innovators the freedom to act upon their ideas.

Issue 3

Problem: Certain licenses create opportunities to fork the code base. Consequences include VA Linux, Red Hat Linux, Free BSD, OpenBSD, Net BSD and Mac OSX.

Proposal: Require developers to license enhancements back to author who commits to withhold them from common code base until expiration of the proprietary period.

Issue 6

Problem: Technology markets compete on *components vs systems*. But monolithic sellers lack “best-of-breed” parts while fragmented sellers suffer from multi-party bargaining.

Proposal: Commencement of the free period permits near zero cost transfer of enabling technology throughout developer pool. Best enhancements enter the code base with trivial bargaining cost distortion.

Themes & Takeaways

- The creation of new information is dependent on ability to reuse it – this is information as *process*, which is especially important for SW.
- Information *is* different from tangible goods
 - Nonrivalry permits 0 MC reproduction for network effects.
- Search:
 - Decentralization is a very effective method of parallel search: “all bugs are shallow”
- Negative Info: example is a competitor learning something from an open system it can use against you.
- IP: There is dynamic tension between closed and open systems
 - Open: allows reuse, network effects, and “future proofing” for consumers promoting adoption
 - Closed: allows you to charge more, thwart competitors
- Information product design:
 - A good biz model may not be static but may transition

More information available...

- copyflex.org
- gparker@tulane.edu
- mva@bu.edu, marshall@mit.edu