

2nd National Software Summit (NSS2) — The Software Industry
Software Productivity Consortium - Reston, Virginia - April 27, 2004

The Software Industry and Economic Security

Terry Bollinger
The MITRE Corporation

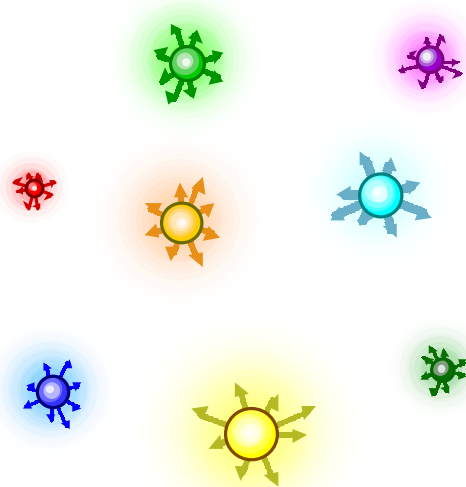
April 27, 2004

Note: *The author's affiliation with The MITRE Corporation is provided here for identification purposes only, and is not intended in any way to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author within this personal presentation.*

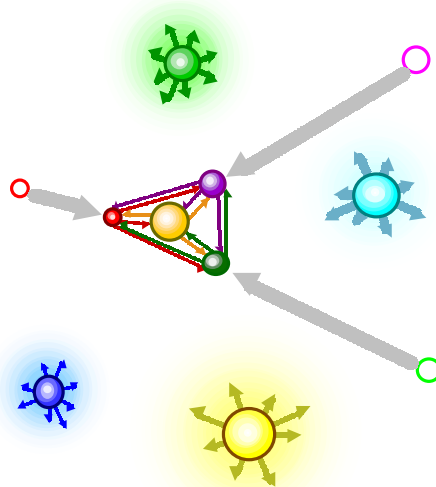
Impact of the Internet on the Software Industry

1970-80s: Era of the Software Firm
(costly data transport drives structure)

Stranded Resources

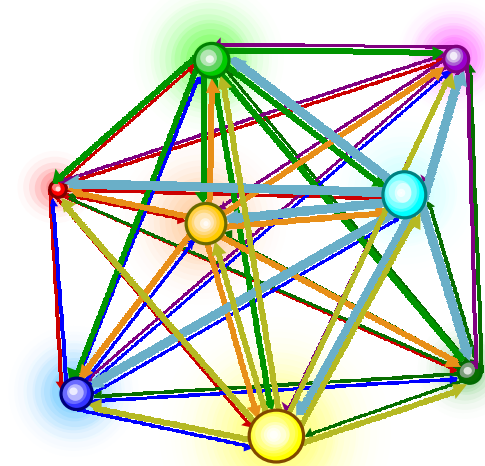


Coase Localization



**RESULT: Innovation is enabled,
but “invisible hand” is limited**

1990s-on: Free Market
(cheap transport dominates)

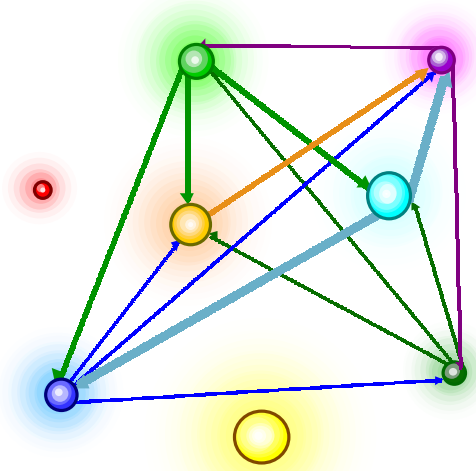


**RESULT: Innovation
accelerates rapidly**

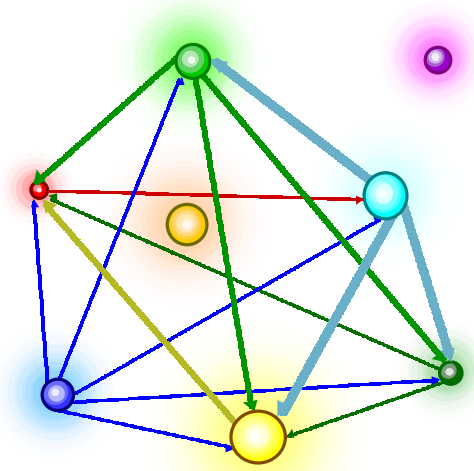
Source: “Software Cooperatives” by Terry Bollinger (terrybollinger@erols.com)

Structural Implications for Software Industry

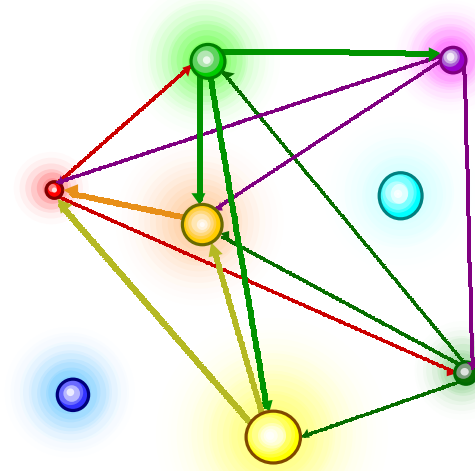
Cooperative A



Cooperative B



Cooperative C



REASON: “self selecting” global subsets retain free market advantages

RESULT: Cooperatives (barter-based) and, eventually, Consortia (fee-based) begin to dominate the market

Ownership Models in Cooperatives

■ **Schoolhouse** (e.g., GPL)

- Jointly & voluntarily built. All may use it, but no one person or group owns it.
- *“Once a schoolhouse, always a schoolhouse”*: Parts may be reused, but only to build more schoolhouses.

■ **Public Service** (e.g., BSD, Artistic)

- Jointly built using voluntary donations, but allows reassignment as private property (e.g., Apple OS X)
- The most popular alternative to the GPL License

■ **Liberal Lease** (e.g., LGPL)

- Parts remain “property of the school,” but can be freely reused to enhance the value of private property
- Popular with small businesses that rely on open source

What About Traditional Software Firms?

■ The profit incentive remains intact!

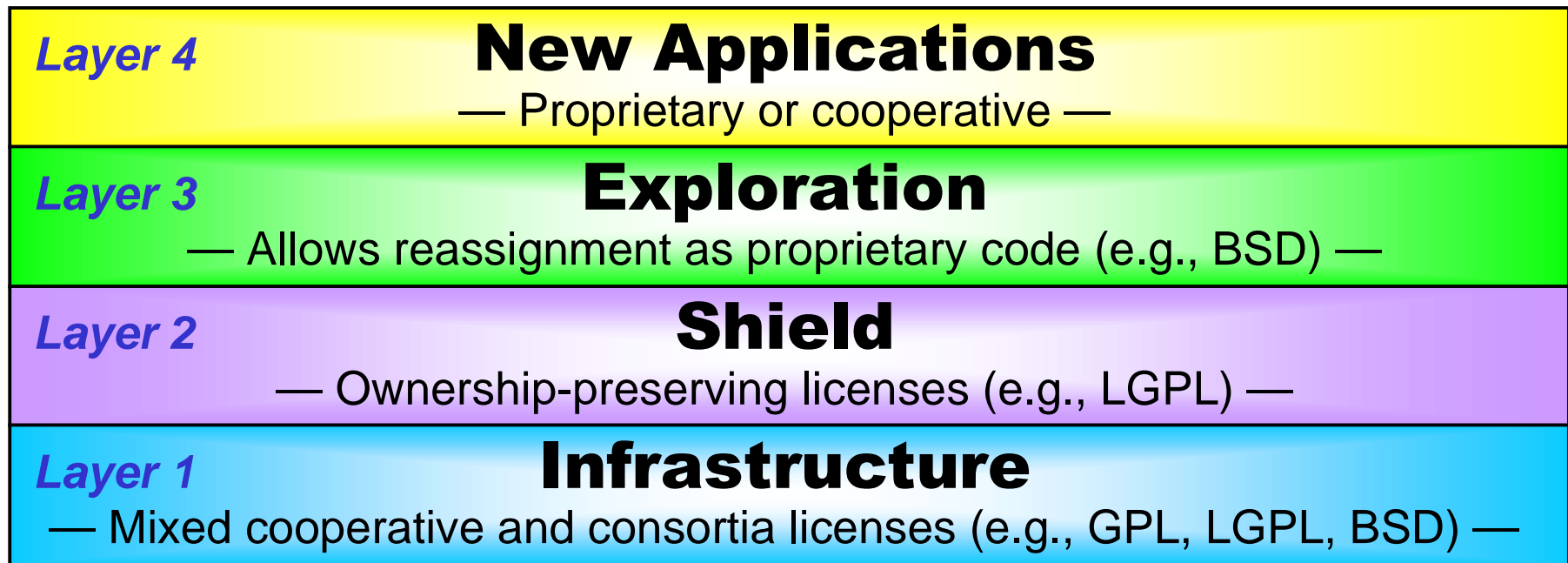
- Consortia “flatten the playing field” ...
- ... but they do *not* remove classic profit incentives
- Ironically, companies that refuse to use consortia are the ones most likely to suffer competitively:
 - Coase-localized (traditional) software companies cannot easily compete with free-market consortia working the same problem
 - Lack of participation in global consortia limits employee abilities to understand and apply viable low-cost consortium options

■ Refocusing and restructuring is needed

- The “refined gold” model of software company structure:
 - Maximize use of, and participation in, consortia
 - Discourage re-creation of consortium-supported software
 - Focus non-shared work and creativity primarily on difficult, unique, and high-payoff innovations — the “refined gold”

Example of a “Refined Gold” Architecture

New Applications: Software that is unexpected or solves a hard problem



Infrastructure: Software whose value increases as it is more widely shared

Mutual Software Trust (MST)

■ The problem:

- When groups with varying level of trust of each other must work together, how can they share infrastructure?

■ A lesson from history:

- The simple handshake developed first as a way of proving that neither side was carrying a weapon
- For software, similar “open inspection” principles apply

■ A partial solution: **Mutual Software Trust**

- MST means all shared software resources must be fully exposed for inspection at any time, by any participant
- MST is gradually **driving internationally shared software more and more towards open source (only)**