

## IVb. From Code to Product

- Testing and Quality Assurance
- Product Enhancements
- Manufacturing
- Customer Support
- Training
- Intellectual Property Protection for Software
- Open Source Software

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.1

## Software Testing and Quality Assurance

- *Software product quality must be systematically tested to guarantee that it meets desired standards; this includes alpha testing within one's own organization and beta testing at carefully selected customer sites. (#27)*

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.2

## The need for testing

- Cost of fixing bugs grows dramatically, e.g.,
  - If 1 in design phase, perhaps 10 in implementation phase, perhaps 100 in the testing phase, perhaps 1,000-100,000 after shipment
    - WindowsNT — 4M lines of code —100 testers
    - Windows95 over 10M lines of code
    - WindowsXP — released with over 50K bugs
- The “just one trivial fix” syndrome — “just a simple matter of programming”
- Serious bugs appear bizarre and illogical

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.3

## Testing and quality assurance

- Quality assurance (QA)
  - The systematic testing of product quality to guarantee that it meets desired standards
- Test suites — systematic, comprehensive sets of testing programs
- The use of test suites in quality assurance
  - Running test suites manually
  - Automated test suites
- Locating QA in marketing rather than development or sales

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.4

## Alpha testing

- Final testing within your own organization
- Testers not part of the development team
- How much is enough? Want to be very solid before proceeding to beta test

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.5

## Beta testing

- Next stage of testing, realistic product shakedown, at cooperative customer sites
- Goals: bugs, performance data, comfort, reference accounts
- Criteria for beta site selection: eagerness; expertise; prestige; commitment for cooperation, feedback, reference account
- Importance of support

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.6

## Software Product Enhancements

- *To remain competitive, convince customers that the company is innovative, and provide an additional source of revenue, software must be continually improved through a carefully managed process of enhancement, testing, and new releases. Yet what one omits from new releases is even more important than what one includes. (#28)*

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.7

## Product enhancement process

- Adding new features and fixing bugs
  - Gathering, organizing, prioritizing wish list
  - Choosing what goes in and what does not
- Managing the release process
  - Adhering to plan: it's only a "trivial change"
  - Quality assurance of new releases
- The dangers of version proliferation
  - Example: Property management software by Minicom and J&E — J&E had versions for each customer! (and died, now part of Geac)

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.8

## Software Manufacturing

- Usually outsourced
- Internet distribution
  - Shipping demo copies
  - Shipping production copies
  - Shipping bug fixes and updates
  - Documentation and support available on Web

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.9

## Customer Support

- *Competent and responsive customer support is essential in today's software marketplace. It provides additional revenue, and is required for effective sales to customers who are not computer specialists. It is a method of corporate and product differentiation, and a source of intelligence about your product and those of your competitors. (#29)*
- Example: WordPerfect support of 3000 — Corel reduced to <500 full-time support staff

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.10

## The importance of support

- Complexity of software and systems
- End user customers not computer specialists
- Customers in distribution channel (e.g., dealers) overloaded with multiple inconsistent systems

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.11

## The difficulty of support

- Differences in backgrounds and capabilities
- Lack of a common language
- Describing human-computer interactions at a distance
- Multiple interfaces, versions, & environments
- Psychological drain of being a full-time helper

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.12

## Methods of support

- Organizational
  - Dedicated support staff
  - Rotating staff, including management and technical staff (NO exceptions!)
  - Hierarchy of expertise
  - Ensuring followup — support logs, databases

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.13

## Methods of support (cont'd)

- Technical
  - Telephone
  - Fax
  - Email
  - Web database of FAQs, updates, problem solutions, bug fixes (patches)
  - Direct machine connections
  - Just-in-time videos

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.14

## Learning from support

- Bugs in the program
- Features needed in the program
- Problems with the user interface
- Problems with the documentation
- What the customers really love
- What the competitors are offering
- What the competitors really have
- The key — Getting this information to marketing and development!

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.15

## Costing and pricing of support

- Costing
  - Time of support staff, backup developers, etc.
- Pricing
  - Warranty period — typically 90-180 days
  - Support period — typically annual
  - For mainframe, mini, workstation products — typically 10-25% of list price per annum
  - For micro products — sometimes toll-free hot line, sometimes customer pays phone charges
  - For micro products — no cost, or charges on the order of \$2/minute, \$25/incident, \$300/year and up

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.16



## Training

- *Functionality and complexity of software seems to grow at least as fast as its alleged “user friendliness,” hence customer training is essential and can serve as an additional source of revenue. Corporate training skills can also be applied to the professional development and internal training of one's own employees, which is an essential component of honing and sharpening one's competitive edge. (#30)*

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.17

## Training one's customers

- The importance of training
  - Complexity of software and systems
  - End user customers who are not computer specialists
  - Training gets them to the point where support is useful
- Training as a source of customer satisfaction and product differentiation
- Training as a profit center

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.18

## Training one's employees

- Rate of change within the computer and software industry
  - New paradigms and new methods
  - Typically every 3-7 years
- Advantages of employee training
  - More able staff
  - More motivated staff

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.19

## Intellectual Property Protection for Software

- *Although the intellectual property of software should be protected, with the help of a knowledgeable attorney, through trade secret, copyright, patent, trademark, and contract law, speed and agility is the best protection for a new software firm. (#31)*

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.20

## Preventing software piracy

- Dimensions of the problem
  - Most serious with micros, personal use, esp. games
  - Estimates range from 25% to 95%
  - Reduced profitability, even bankruptcy
- Protections
  - Copy protection, but some disadvantages
  - Physical: delivery in ROM, external key devices
  - Electronic: encryption, passwords, usage restrictions
  - Human: providing valuable, truly needed support
  - Legal

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.21

## Safeguarding software intellectual property

- Trade secret law
  - Protects unique and secret aspects of an idea
- Copyright law
  - Protects the “written expression” of the idea
- Patent law
  - Gives owner the exclusive right to make, use, or sell articles that embody the idea

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.22

## Safeguarding software intellectual property (cont'd)

- Trademark law
  - Protects marks intended to identify the good or service or the source of the good or service
- Contract law
  - Protects whatever is agreed to in a contract

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.23

## Trade secrets

- Information must be a business secret, not generally available
- Information must have commercial value
- Owners must take “reasonable steps” to inhibit unrestricted disclosure, e.g., via employee nondisclosure agreements, machine passwords
- Used to protect ideas, concepts, processes, business plans, source code, etc.

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.24

## Trade secrets (cont'd)

- Source of difficulty: Discussions of possible partnerships, joint ventures
  - Solution: Selective disclosure based on need to know, like peeling back the layers of an onion
- Recently, \$1M awarded in B.C. for theft of source code by company commissioning software

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.25

## Copyrights

- Originally intended for writing, works of art, etc.
- Restricts reproduction, distribution, performance, display of copyrighted work
- Restricts preparation of derivative works
- Used to protect source code and object code
- Cost is low, but level of protection is also low, as it doesn't deter reimplementations
- Recent movement to protect "look and feel" of software likely has failed
  - Lotus (successful), Apple (unsuccessful)

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.26

# Patents

- Originally intended for processes, typically embodied in “machinery”
- Not applicable to mathematical formulae
- Recently, widespread and successful use to protect software processes and designs
  - 1200 software patents issued in U.S. in 1990-91 (300 in 1988-89)
  - The number is now likely 75,000-100,000 per year
  - Many of these are “bad” patents

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.27

# Patents (cont'd)

- Level of protection is high, but obtaining it is costly (more than \$10K) and time-consuming, and trade secrets must be revealed to obtain a patent
- Even costlier to defend (more than \$100K), yet payoff can be great (\$120M from Microsoft to Stac Electronics)
- “Offensive” and “defensive” uses of patents

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.28

## The source of confusion

- Copyrights are for writing
- Patents are for processes
- Software is the first artifact that is both writing and process

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.29

## Open Source Software

- *An increasingly interesting alternative to “traditional” means of software protection is to publish on the Internet the source code for your technology. This is typically termed releasing your software “open source”. Open source advocates cite reliability, speed of bug fixing, performance, adaptability, scalability, and security as benefits. Open source software also often gains favorable publicity and mindshare. (#32)*

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.30

## Definition of Open Source

- “Open source promotes software reliability and quality by supporting independent peer review and rapid evolution of source code. To be OSI certified, the software must be distributed under a license that guarantees the right to read, redistribute, modify, and use the software [source code] freely. ”  
<http://www.opensource.org/advocacy/faq.html>

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.31

## Examples

- Linux (GNU/Linux)
- Apache Web server
- Samba Windows-client file/print server for UNIX
- Perl programming/scripting language
- Mozilla web browser from Netscape
- Darwin release of Apple Mac OS X

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.32



## Business Models

- Sell Support
  - Integrate, brand, distribute, service, e.g., Red Hat
- Loss Leader
  - Give some software away as market positioner for selling closed software, e.g., Netscape
- “Widget Frosting”
  - Hardware company w. open source software seeks better/cheaper drivers and utilities, e.g., SGI Samba
- Accessorizing
  - Selling books, hardware, complete systems with open source software pre-installed, e.g., O’Reilly

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.33

## Issues

- Success stories usually associated with “miracles” (e.g., Linux), established mission-critical software (e.g., Apache), server software
- Legal issues still complicated (30 different licenses on [www.opensource.org](http://www.opensource.org))
- Applicability to startups unclear
  - Still need lots of marketing and promotion \$
  - Need a “winner” to build a community of contributors
  - Need a “winner” to build revenue stream
  - Will the “movement” be sustainable over time?

---

The Business of Software Jan.-April 2003 ©1992-2003, Ronald M. Baecker Slide 4b.34