

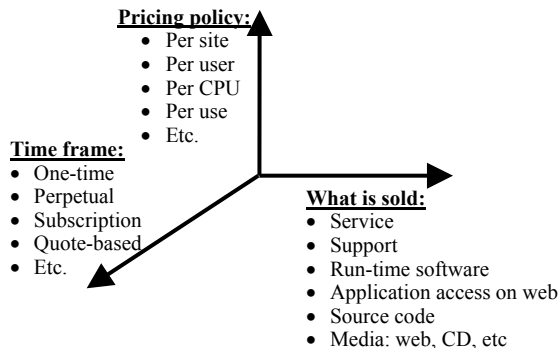
# SOFTWARE BUSINESS MODELS

by Robin Steele, Renaissance Executive Forums and Bert Vermeulen, Corp21

Creating a software product or internet application is relatively easy. Making this a viable, profitable, and sustainable business is challenging. The internet creates new ways to promote, distribute and support software. It also lowers entry barriers. This paper presents concepts and ideas to help you determine the best model for your software business.

## Categorizing software business models

For the purposes of this paper, software can be defined as: *"Intellectual property that can be licensed along three dimensions: (1) what is sold, (2) pricing policy, and (3) time frame."* This is further detailed in the following diagram:



Here are some examples of software businesses characterized using this framework:

- **Traditional PC products:**
  - Run-time software is licensed per CPU.
  - Software purchased with a one-time fee.
  - Limited support bundled with the software.
  - Software delivered on shrink-wrapped CD purchased through the dealer channel.
  - No additional services offered.
- **Enterprise client-server applications:**
  - Run-time software is licensed per site or per CPU based on the number of users.
  - One-time fee for the software plus a separate subscription for support and upgrades.
  - Software delivered on a shrink-wrapped CD sold through direct channels.
  - Consulting offered on per hour basis.
- **Web-based applications sold using the ASP (application service provider) model:**
  - Initial service to setup and customize the application. Priced on a one-time custom basis.
  - Support bundled with subscription fee.
  - Delivered as a hosted web application. Priced as a subscription based on number of users.

Other popular models are the custom software/consulting model and various open source models. Mixing and matching gives many possible options.

Companies often offer multiple types of licenses. This table (from Software Magazine June/July 2001) shows the percent of the top 500 software companies using each of the following pricing policies:

Pricing policy	% using
Enterprise license	61.0%
Site license	51.4%
Per concurrent user	42.0%
Subscription	24.8%
Leasing	21.8%
Per node	16.8%
Per server	16.8%
Per connection	9.6%
Metered usage	7.6%
MIPS	5.6%
Per session	4.0%
Other	14.4%

Software Magazine (same issue) gave the following breakdown of support offered by these firms:

Support method	% using
Telephone	80.8%
Email	78.8%
Web page	70.4%
On-site technician	62.4%
FAX	59.0%
CD-ROM	32.4%
Bulletin board service	21.4%
Other	11.8%

Another decision is what service and support is free and what generates revenues. General guidelines are:

- Things done to support or generate sales are free:
  - Pre-sales support
  - Guidance on how to use resources like manuals.
  - Installation assistance.
- Post-sales support is typically paid. Examples are:
  - Education and training.
  - Software questions and configuration help.
- Customization and consulting is generally paid:
  - General consulting on business processes.
  - Custom software solutions.

### Software business economics

Software companies have a cost structure that is different from hardware related businesses:

- Manufacturing cost is zero or close to zero.
- Product development costs vary dramatically depending upon the type of product and a company's competitive position: Examples are:
  - Products like Microsoft Windows & Microsoft Office have R&D cost less than 10% of revenues and probably less than 5% of revenues (Source: analysis of Microsoft 2003 10K) due to the high volumes in which these products are sold and Microsoft's dominant market position.
  - Software consulting businesses and developers of custom software have almost all of their costs as "development" costs.
  - The majority of software businesses have R&D costs somewhere between these extremes, typically between 8-30% (Source Software Magazine June/July 2001).
- Except for custom software or consulting businesses, marketing is typically the largest single element of a software company's cost structure. This is because it takes a lot of effort to rise above the "noise" of the thousands of software products that are offered (due to low product development and marketing costs).

### Pricing

When determining how to price each element of your business, it is important to look at this from the perspective of your business and from that of the customer. From your perspective, you should ask:

- Which customers are our core business?
- With whom do we compete?
- What are the five most important benefits our target customers seek?
- How will we make money in our business?
- What are our avenues for growth and expansion?

When doing an analysis of how to price each element of the product, service and support offering, it is important to keep in mind that the customer may perceive value in three ways:

- Functional: What can the product do for me?
- Economic: How does it save time and money?
- Emotional: How does it connect with me? Cars are a typical example of a product that is sold on an emotional level. All cars get you to the destination about the same speed for the same cost, yet there are huge differences in how successful a car is based on things like its styling.

It is important to think of the value of your offering broadly and to keep in mind that:

$$\text{Value} = \text{Consumption Value} + \text{Transaction Value}$$

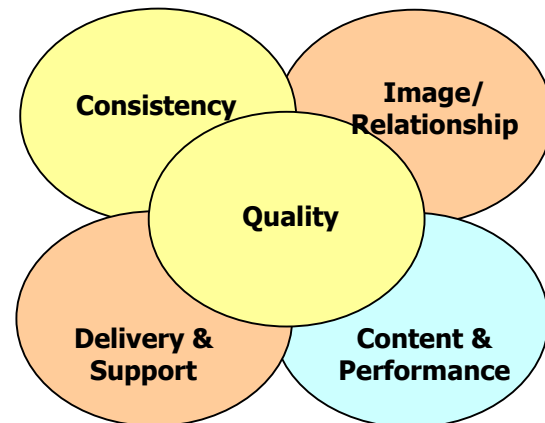
Where:

**Value** is the total value of the product

**Consumption Value** is the benefit of the product to the user while he is using it.

**Transaction Value** is the benefit of the product to other parts of the organization.

It is also important to remember that the "product" has multiple interrelated elements:



Taking these things into consideration, the most common pricing methods are:

- Cost-based: Cost + x%
- Good value: Based on price/performance.
- Meet the competition's pricing.
- What the market will bear. Skim highest value customers who will pay the highest premium.
- Penetration. Price to gain market share.
- Follow the industry pricing leader.

A final consideration is that there are certain price points (\$100, \$200, \$500, \$1000) which, if you can stay below, will make it possible for a lower-level decision to be made in most organizations.

### The software ecosystem

Another useful tool is to create a diagram of your software ecosystem. This should take into account the key phases of the customer buying cycle: awareness, evaluation, ordering, delivery, deployment, training, support, renewal, and obsolescence and the importance of and status of various processes to meet market needs. The diagram at the top of the next page is a simplified example for a low-touch, low cost software business:

Awareness	Evaluate	Order	Deliver	Deploy	Train	Support	Renew	Obsolete
Product info easy to find on web (High importance, Works well)	Product info easy to find on web (High importance, Works well)	Product on web with a credit card (High importance, Works well)	Secure web download (High importance, Works well)	Single-user version (High importance, Works well)	Manuals on the web (High importance, Works well)	Phone support 9-5 Pacific time (High importance, Works well)	Annual support renewal (High importance, Works well)	Product migration plan (Medium importance, Works well)
Web seminars & trade shows (High importance, Works well)	Low cost sales channel (High importance, Works well)	Telesales (High importance, Works well)	CD mailed Within 24 hours (High importance, Works well)	Network version (High importance, Works well)	Help menus with the product (High importance, Works well)	24-hour phone support (High importance, Works well)	Product upgrade marketing (High importance, Works well)	Support plan for old versions (Medium importance, Works well)
Guerilla postings on bulletin bds (High importance, Works well)	Links to software partners (High importance, Works well)	Monthly subscription (High importance, Works well)	Available in retail channel (Low importance, No system)	Web-based ASP version (High importance, Works well)	One-hour phone-based (Medium importance, Works well)	Live support in Europe (High importance, Works well)	Customer web renewal system (High importance, Works well)	Open source EOL strategy (Medium importance, Works well)
Advertising (High importance, Works well)	Interactive demos on website (High importance, Works well)	Direct sales (Low importance, No system)		OEM license (Low importance, No system)	Web-based (Medium importance, Works well)	Email support (High importance, Works well)	Licensee database (High importance, Works well)	
Lead generation programs (High importance, Works well)	Comparison tools on website (High importance, Works well)			Enterprise site license (Low importance, No system)	Face to face (Low importance, No system)			
Customer registers for special info (Medium importance, Works well)	Call center qualifies prospects (Medium importance, Works well)			Version dongle for security (Low importance, No system)				
Bulletin board and community (Low importance, No system)								

Legend:

- High importance
- Medium importance
- Low importance
- Works well
- Partial implementation
- No system

Sample low-cost, low-touch software ecosystem

The objective is have an integrated software business model in which:

- The combination of product, service and support has the highest perceived value to the customer, the lowest perceived entry-level cost.
- Revenue to the provider is maximized by generating it in as many parts of the product, service and support offering as possible.
- Low touch/low cost (web/telesales) delivery methods for software, service and support are used wherever possible and appropriate.

The authors of this article can help with this process.

**Conclusion**

We hope that you have found this brief helpful. Please let us know if you have any questions or suggestions for areas that should be covered in more depth. If you need help, with the business model for your business please contact us.

The following references may be helpful if you'd like to explore this topic in greater depth on your own:

- Fink, Martin. *The Business and Economics of Linux and Open Source*. (c) 2003 Hewlett-

Packard Company. Published by Pearson Education/Prentice Hall PTR.

- Fishman, Steven. *Web and Software Development, a Legal Guide*. (c) 1994-2002 by Steven Fishman. Published by Nolo Press.

**About the Authors**

Robin Steele is president and owner of Renaissance Executive Forums in Northern Colorado. Robin has over 20 years of experience running software businesses, managing R&D organizations, and providing consulting services. She now provides advisory board services to CEOs and business owners. Robin is located in Fort Collins, Colorado. For more information, see: <http://www.executiveforums.com/locationsList.asp?loc=6>. Robin also has a more detailed Software Business Models presentation which you can obtain directly by contacting her through the website above.

Bert Vermeulen founded Corp21, a company that provides advisory and incubation services to business owners, general managers, inventors and other entrepreneurs around the world. For more information, see <http://www.corp21.com>.