

Key Findings

September 1, 2004

The Emerging New Enterprise Software Industry Business Model

Even in a slow week for CRM news you might have missed a couple of announcements that may have a significant impact on the future of the enterprise software market, including CRM. Last week RightNow, a newly public OnDemand call center company announced that it had joined the WS-I an independent industry standards group promoting Web Services. RightNow is not alone, there are 170 software companies already involved in the consortium. At the same time WS-I announced general availability of its Basic Profile 1.1, Attachments Profile 1.0, and Simple SOAP Binding Profile — Web Services standards.

Web Services is a set of technology standards that help disparate applications share information using XML technology and protocols like SOAP. The Web site for the World Wide Web Consortium, www.W3C.org, describes SOAP as “[A] lightweight protocol intended for exchanging structured information in a decentralized, distributed environment.” And further on, the same site says, “The advent of XML makes it easier for systems in different environments to exchange information. The universality of XML makes it a very attractive way to communicate information between programs. Programmers can use different operating systems, programming languages, etc., and have their software communicate with each other in an interoperable manner.”

Interoperability and Web Services

Some readers will no doubt look at this and think I am reciting ancient history. While they’re not exactly ancient, XML, SOAP and other protocols have been around for a while and many people — especially those who write code for CRM and other enterprise applications — know of these technologies and increasingly refer to them under an umbrella term ‘Web Services’.

What’s important about Web Services is that they enable great interoperability and extensibility in applications and they can be combined in a loosely coupled way that can help users achieve complex operations. Almost like Chaos theory, simple services can interact with each other to deliver complex and sophisticated results.

The business impact

If you are like me, knowing something about the technology is nice but understanding the business impact is even more interesting. It might be hard to overstate the potential business impact of Web Services because they represent the key to a very old lock in the enterprise software industry.

For the whole history of the software industry applications barely spoke to each other and then only with great difficulty and expense. Interfaces were always in need of tuning as one side of the integration or the other changed to meet evolving business needs. Needless to say integration was brittle.

But Web Services have the potential to change all that, and with it, the prevailing business model of the enterprise software industry will change also. What might a new or at least the next software industry business model look like? For starters, like the applications themselves, it will probably be much more decentralized. Think of this: virtually every industry we know from auto manufacturing to movie production has undergone a decentralization process over the last few decades, except enterprise business software.

The enterprise software industry is still vertically integrated to a very high degree. A vendor is responsible for research and development, sales, marketing, and a hundred other things that go into delivering a product to the market. But what if, rather than developing the world's umpteenth GL package, a software company could simply link in one that's already built with minimal fuss?

A business model for the 21st century

The mainstreaming of Web Services is important because it promises to change the whole business model of the enterprise software industry by enabling it to deverticalize. Some changes you can expect include:

- A two tiered industry made up of developers and publishers. Developers do what they've always done — and less. The publishers take over responsibility for sales, marketing, and some support functions.
- Greater reliance on a hosted delivery environment. Publishers will support many thousands of users of standard applications resulting in an automatic market that additional developers will want to tap into.
- Software revenues will become much more predictable as a result. Revenue predictability is just now being tested by Wall Street and companies like Salesforce.com and RightNow. But as a practical matter, who wouldn't like to have fewer earnings calls that start with "Oops!?" They don't call it the "utility model" for nothing.
- Reduced cost of entry for new software companies because their primary objective will be to build relevant, working applications that conform to integration standards. The publisher will be responsible for the costly sales and marketing effort.
- Because the publisher will have a relatively captive customer base (similar to your phone or electric company) the publisher will have an instant sales channel for new products and the publisher will be able to sell through the channel at a far lower cost than today's software companies encounter.

Many people have noted the seeming lack of some “new, new thing” to propel the tech industry. Some have advocated taking care of the unfinished business of making everything that we already have work. But making everything work doesn’t mean hanging on to the status quo. The new, new thing is a new business model based on an architecture that really can promote interoperability. That’s a worthwhile challenge and one that will employ quite a few of us over the next decade.

###

