

# PLATFORM AS A SERVICE

*A Beagle Research Group ViewPoint™*

*Published September 2007*

## VIEWPOINT: PLATFORM AS A SERVICE

---

### Abstract

The software development tools market is littered with obsolete or dead-end products that attempted to make incremental improvements in the way enterprise software is created and maintained. Although a few improvements succeeded spectacularly well, such as the relational database (RDBMS) and SQL, most approaches to streamlining the development of procedural code proved to be ineffective. A long list of partial solutions starting with CASE tools and continuing to the present day with solutions like .Net all suffer from many of the same shortcomings — they are code-centric, they work on a limited number of hardware platforms, they are expensive to purchase and use (especially factoring in the cost of technical expertise) and they require a bewildering array of supporting infrastructure that is exposed to, and must be managed by, the user. In a word, they are complex and their complexity is built into the fabric of these approaches making it impossible for users to factor complexity out.

The antithesis of complexity is not simplicity but modularity. Modularity enables users to sub-divide large tasks into smaller efforts that may be attacked by specialists and the standards provided by modularity give all who participate in the development process a blueprint for how to connect their work with the work of any other people with a similar mission, not matter if those people are down the hall or half way around the world. Together modularity and standards set the stage for a new paradigm in application development, but by themselves, they are not enough. There are modular, standards based approaches to code development on the market today but they have not delivered an application development paradigm that transcends the limitations of building code.

The evolution of the software development platform has provided the last necessary building block that users need to retire the old paradigm of procedural code development. The platform gives purpose to modularity and standards and provides a management layer that shields the user from concerns about operating systems, databases, security, networking, resource provisioning and much more. Just as today's car drivers do not need to have a working knowledge of transmissions or fuel injection systems, today's application developers no longer require detailed knowledge of their back-end infrastructures.

Salesforce.com's recently introduced force.com appears to be a prototypical example of an on-demand application development platform which supplies solutions for many of the requirements outlined here. Force.com is qualitatively different from similar tools that are intended to support the customization and integration needs of application users supplying the wherewithal for users to build new applications from the ground up.

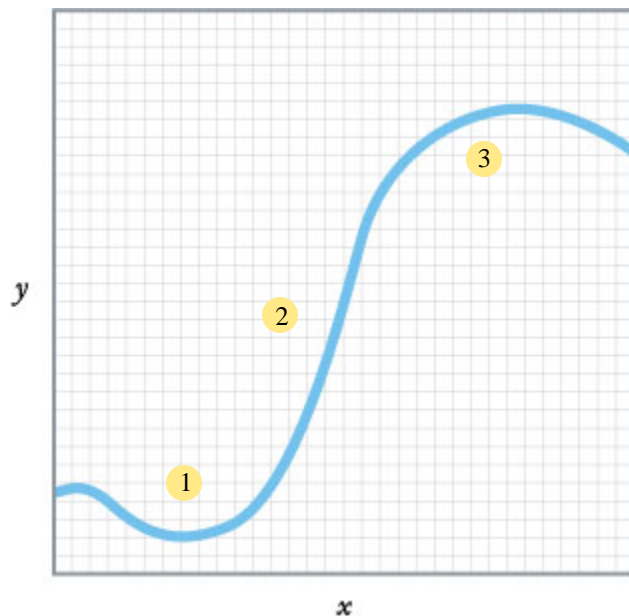
Importantly, providing a reliable platform that shielded drivers from the complexity of cars resulted in enabling more people to drive which, in turn, created our modern mobile society. Similarly, shielding those who need to make business applications from the complexity of computing systems will unleash a new round of creativity in business applications and result in a new level of sophistication and awareness in commerce that enables companies to be more intuitive and responsive to the world around them.

## End of the road

*The application software development paradigm is nearing the end of its useful life.*

Every entity has a life-cycle, be it a hamster or a product, which charts a trajectory that can be best drawn by a sigmoid or S-curve (Figure 1). The S-curve itself can be divided into three parts which include inception and a period of immaturity (1), a productive and robust mature phase (2) and finally senescence (3) when it must be replaced. The software application paradigm has followed this curve starting with assembly language followed by compilers and, in the last decades, by a long list of code generators, libraries, “frameworks” and other products chiefly designed to extend the useful life of the code development paradigm.

Figure 1. The sigmoid curve



Source: Beagle Research Group, LLC September 2007

Today, the traditional software paradigm is in serious trouble. Application development has become complex and expensive to the point that it presents a serious impediment to business growth. Software is expensive and development projects take a long time so that only the most seriously needed applications get bought or built leaving many users to muddle along with paper files or spreadsheet based “applications” that are quickly overwhelmed.

Platform as a Service (PaaS) is being positioned as a solution to many of the bottlenecks of traditional programming. For most tasks business users can build applications

## VIEWPOINT: PLATFORM AS A SERVICE

instead of using spreadsheets, but with the rigor of professionally developed applications. Also, professional developers will find in PaaS solutions such as force.com from salesforce.com analogues to all of the tools they require for modern application development.

### **Economic drivers**

The code development paradigm is not the only technology paradigm to be nearing the end of its useful life. Advances in computer hardware have for decades been the engine of increasing business productivity, extending business' reach and enabling economic expansion.

For much of the last half of the twentieth century advances in computer hardware were enough to propel businesses by enabling them to manage ever greater volumes of data. Moore's Law virtually guaranteed that every 18 months computing power would double and prices would contract thus ensuring that business growth could continue unabated. However, Moore's Law has charted its own S-curve and while computer hardware performance is no longer doubling every 18 months, the demand for ever more powerful and sophisticated business systems continues unabated. New demands caused by globalization, security, governance and other requirements mandate increasingly sophisticated business processes networked across buildings, regions and around the world.

The combined obsolescence of the software coding paradigm and the physical limitations on our ability to increase raw computing power result in a "perfect storm" in which new technology leadership must be invented and deployed to enable global business to continue expanding.

The next technology paradigm to drive the business world will involve the on-demand software development platform. The platform simplifies the many aspects of deployment and maintenance while enabling users — both professional developers and business users — to build increasingly sophisticated business systems. The platform can be used to build in complex business rules that span departments and disciplines; it can enable better and faster integration of disparate systems; and most importantly, it can drive down the costs associated with business software.

### **Modularity and the platform**

The need for modularity and standards has driven the evolution of the application development platform giving users a new and better way to think about and develop business applications.

### ***Technology impacts***

The platform provides an open architecture that has far reaching impacts on applications as well as business.

1. Platform oriented application development and maintenance enables a best of breed approach for supporting end-to-end business processes and models build in a platform can be reused many times to support changing business requirements. The platform also opens up new niches for third party software

## VIEWPOINT: PLATFORM AS A SERVICE

developers and reduces the lag time between application conception and delivery.

2. The platform removes operating systems, databases and other infrastructure from consideration for most application development issues resulting in a simpler environment that is easier for an individual to understand.
3. At the same time, modularity enables multiple developers to work asynchronously to support business processes that span departments and locations.

The software requirements of a typical medium-size business today are more numerous than any single software supplier can provide. Using a software development platform approach enables multiple developers or vendors to build solutions using the same standards for data, user interface, processes and when necessary, procedural code. As a result solutions built on the same platform can then be merged or further customized to meet the end-to-end processing requirements of the target audience. Also, since no single entity has all the best ideas, in a platform based environment multi-party solutions can become the norm rather than the exception. Subject matter experts from many different places can add their value without worrying about whether or not a solution will mesh with the rest of a suite because all applications, either developed or purchased are based on the same platform standards. A best of breed approach can result in more powerful and complex global business processes made possible by modular applications.

### ***Business impacts***

The standards imposed by a modular platform development environment extend the marketplace for many applications with the result that prices can fall. For example, there is a class of orphan applications that exist in every organization that support small number of users. Although these applications are important to the business, they are frequently left to be administered in spreadsheets because they are not important enough to involve IT in their development. Moreover, because these applications are so thinly subscribed to, it makes little economic sense to commercial software developers to build and sell them via conventional means. However, in an organization that bases its IT at least in part on a platform, users can just as easily build these applications using the platform as they could using a spreadsheet. At the same time, some orphan applications can create new revenue centers for software developers who are able to aggregate market demand and larger markets help drive down the unit cost of individual applications.

There will always be a need for professional software developers and vendors though their rolls are changing. As this discussion implies, when the cost of application development goes down, additional profitable software niches open up. In the last few years the industry has seen numerous niches that were once virtually the exclusive domain of spread sheets — such as on-demand compensation management — open up. New niches such as search engine keyword marketing have also opened and the promise of quick and reliable integration driven by platform technology has added a new term, “mash-up” to the IT vocabulary.

Less visible but of even greater importance is the shift that many software vendors are making to providing additional services based on domain expertise rather than sys-

## VIEWPOINT: PLATFORM AS A SERVICE

---

tems implementation and maintenance knowledge. In the sales and marketing vendors are offering more methodology based assistance to help their customers maximize their investments.

### Summary

The S-curve that describes the life-cycle of the software development platform is the logical successor to Moore's Law. Hardware performance was once enough to improve the business performance when manual business processes were first being automated. With most business processes now automated to some degree, additional improvements will come from building more intelligence into applications.

The task of building business applications has exceeded the relatively primitive tools that were first invented for the job. Using coding techniques to build business applications today is like asking a blacksmith to build a fighter jet. We believe that the evolution of PaaS represents an important advance for business software developers and users because it adds a level of precision and reliability to all levels of the lifecycle.

The applications that will run global business processes now and in the future must be adaptable to changing business needs, capable of being integrated quickly and maintained efficiently and they must be impervious to the failure points inherent with hand crafted code. These requirements call for a level of standardization and automation that many people have sought out for several decades. After numerous dead ends, the on-demand development platform appears to be the logical solution.

## ABOUT BEAGLE RESEARCH GROUP

*Beagle Research Group is a consulting and market research organization focused on emerging technologies and companies that will have an important impact on the way business is conducted in the years ahead. Our work is based on professional standards of quantitative and qualitative research which informs all of our publications.*



**264 Greenbrook Drive  
Stoughton, MA 02072  
781-297-0066**

**[www.BeagleResearch.com](http://www.BeagleResearch.com)**