Is a Dominant Service-Centric Sector Good for Diversity of Provision?

John Hutchinson, Pete Sawyer, James Walkerdine
Computing Department, Lancaster University, Lancaster LA1 4YR, UK
{hutchinj, sawyer, walks}@comp.lancs.ac.uk

Abstract

An obvious assumption underpinning the immense interest in service-oriented computing is that it is an inherently Good Thing, by which we mean that robust processes and tools for developing service-based systems will bring benefits for service providers and service consumers. The arguments, in terms of consumer choice and flexibility, are certainly quite convincing. However, in this position paper, we question the nature of the underlying assumption, in a world where requirements are as many and varied as potential users and ask if safeguards are needed to ensure that diversity of provision is maintained.

Keywords: Services, SOC, market, provision.

1. Introduction

Service-centric computing (SOC) is the focus of a great deal of interest. Service technologies offer an excellent abstraction of software systems in business domains, allowing business terminology and processes to drive system modelling and design. In other words, businesses provide services to their customers, and service-based systems help them achieve that. However, adoption of the service ethos, particularly the notions of the service marketplace and the consumption of 3rd party service, require a major change in expectations from those that have pervaded software engineering thus far. The consumption of externally provided services assumes that consumers will be able to satisfy their requirements using available services, rather than by specifying their requirements and having systems built to satisfy them. As SOC is relatively immature and a good deal of attention is paid to the development of services themselves, this assumption is easy to overlook, but many of the proposed benefits of SOC assume the 3rd party provision model, and therefore inherently make this assumption. This paper asks if this assumption is valid, given that individual users, or consumers, have an infinite variety of requirements and that "taking what is on offer" is merely a pragmatic compromise when no alternative exists.

The paper is organised as follows: section 2 considers the nature of and benefits of off-the-shelf (OTS) services; section 3 considers the dual nature of standardization and section 4 provides a few concluding remarks.

2. The Nature of and Benefits of OTS Services

Whilst not wishing to get embroiled in a service definition loop, it is important to recognise that SOC is dependent on services being of a certain type. Certainly there are technological standards, but services assume a likelihood that certain types of features will be present [9]. However, Szyperski [6] makes an interesting observation that services are not software at all, and explains: "What delivers the service is the software executed by some abstract machine (platform, operating system, virtual machine, …) that, in the end, is grounded in a physical machine. … The entire tower of abstractions, right down to the physical machine, still doesn't deliver a service. It needs to be paired with some operating agent (in the end, a person) to embed the machine into an infrastructure (power and Internet grid, physical enclosure in some building, physical security and so on) and run it." He summarises, "A software service is the pairing of an operating agent and infrastructure with the software itself, implementing the service functionality and offering it through some interface."

So, in establishing three enabling components necessary to deliver some service functionality via some interface, which is itself a working definition of a service (i.e. operating agent, infrastructure, and software), Szyperski reminds us that services are dependent upon system elements against which potential service consumers cannot express their requirements. In essence, this should be the very thing that makes the service abstraction so attractive to businesses, a complete avoidance of concern about *how* a service is delivered, so long as it is the right service. Consider the main benefits of successful component-based development that Clements [2] describes:

- *Reduced development time* – the time to buy a ready-built component is less than that required to design, code, test and document it – "*assuming that the search for a suitable component does not consume inordinate time*" [2] (emphasis added);
- *Increased reliability of systems* – a component used

by many people should be subject to considerably more user-testing than one that is custom developed;

- *Increased flexibility* – if approached properly, the use of bought-in components should bring with it software designed to exploit *any* suitable component, meaning that users can change component supplier to take advantage of better or better priced products.

Although Clements is overtly concerned with components, it is obvious that these benefits should also be associated with OTS services. This excellent summary highlights the *business* and *technical* nature of the benefits: systems should be delivered faster, work better and be more flexible than custom-built software.

However, to achieve these benefits requires that the services should be standardized: a high degree of variability in what is offered threatens to make the selection process too time consuming; if the range or services available is large, then the user base for each will correspondingly shrink and finally, if services are to be truly interchangeable, they must be almost identical in the way they are delivered.

Do these assumptions really promise to deliver the rich diversity of functionality required by end users?

Clements also considers the risks associated with a shift to using software components. (Here, we define *risk* as exposure to the possibility of delay, of economic or financial loss or gain (or even of physical damage or injury) as a consequence of the uncertainty associated with using particular components (adapted from [3])). Again, we can consider these relevant to SOC:

- *Supplier risk* – Purchasing and using a third party component entails "buying into" the company supplying it. Carney [1] stresses that vendor concerns should play as important a role in component selection as component functionality; thus, issues such as the "business obligations" and "financial condition" of the vendor must be considered as a way of addressing supplier risk;

- *Obsolescence risk* – Components are supported according to the business strategy of the developer and/or supplier. A selected component that is integral to a working system may therefore become obsolete because of, ostensibly, unrelated shifts in the supplier's business. Again, Carney [1] suggests that the risk of obsolescence can only be addressed by assuming and preparing for continuous upgrade;

- *Performance risk* – The function of a third party component may not be fully testable until after a decision to use it is taken and then may not perform as expected in unintended contexts of use.

The first two of these risks apply further pressure on consumers' requirements to coalesce around a smaller set of standardized offerings because the business risks associated with failure are too great. Innovative offerings from new providers are immediately disadvantaged because they cannot provide the necessary assurances of the provider's "business concerns". (This is something akin to the old phrase "nobody ever got fired buying IBM…".) The third risk is expressed in a way that makes is inapplicable to SOC, but there is a similar risk with using 3rd party services. In a service market place, the "provider of choice" may choose not to offer the performance or other quality of service (QoS) required by a potential consumer, which again forces users to accept what is on offer rather than necessarily leading to an abundance of choice.

3. Standardization: Creative Constraint or Lowest Common Denominator?

If we accept that the benefits of service consumption within SOC are likely to lead to a standardization of the services on offer, is this a Good Thing? There are good examples of standardization, whilst naturally resulting in the demise of alternative approaches, being generally beneficial. A potentially controversial example was the emergence of the standard PC architecture on the back of IBM's dominance. Other PC architectures quickly became specialist niche products whilst the IBM model became the architecture of choice. This is a little controversial because there are no doubt many who would argue that the loss of diversity had a negative impact on computer development. However, this argument is immaterial to the majority of PC users who have no interest in the technical merits of alternative approaches. Thus, this standardization could be considered a "creative constraint" (for a lyrical discussion of the creative force of constraint, see Hofstadter [5]) leaving competing component manufacturers to develop products that would give them business advantage, and leading to an evolution of the standards resulting in an increase in capability.

Taking this as a metaphor for the development of SOC, the future is very optimistic. After an initial "survival of the fittest" scramble, we can expect domains to establish their standardized architectures, which will act as a framework for providers to compete to offer the functionality required by users. In this model, the pain and risk associated with evolutionary change should go hand-in-hand with readily perceivable benefits to consumers.

There is, though, an alternative scenario. The UK high street illustrates just such an alternative with concerns over the dominance of the major supermarkets [7] forcing smaller retailers out of business. Another, more subtle example comes from the plight of rural Post Offices [4]. In these examples, the range of "user requirements" is

reduced to the right of an agile majority to have access to the lowest price for the most commonly required services. Those with requirements for alternative, perhaps niche, services – and here we are particularly concerned with accessibility - are forced to consider their needs "premium rate" or simply of no interest to the large suppliers.

This is a more complex metaphor to apply to SOC. Internet delivered services are not restricted to geographical locations in the way that high street shops or rural Post Offices are. In this respect, this is a reason to champion such provision. However, diversity is still an issue, as is the effect of its provision: local grocery stores are able to choose suppliers for whom it is not viable to supply national chains. When it comes to attempting to meet some of the more niche requirements (again, we are talking about accessibility), the cost overheads of supplying high street facilities in the UK do affect the major providers [6]. But still, the effect is to limit diversity, because even these increased costs, ultimately a recognition of the overheads associated with minority provision, are set within a context of much wider business goals. In other words, limited provision in one context might be reasonable for a major service provider, even if it is done with limited business success and at increased cost to the consumers, if it helps to limit the ability of other providers more generally.

Questions about the business practices of companies that achieve, or aspire to, global dominance are not new, think for example IBM [10], Microsoft [11] and Wal-Mart [12].

Further applying this model to SOC is less optimistic. Here, the requirements of end users are of interest only to the extent of the "lowest common denominator". There is no interest to satisfy the requirements of all users, or even most users. Service suppliers will supply only those services that deliver the greatest profit for the least effort. Of course, there will exist a market for services that are "premium", but a huge gap will emerge between the services that are available to all and those that are available to only premium rate consumers.

4. Conclusions

Of course, the "conclusion" of this position paper is less a conclusion and more a question: is the move to globally accessible web-based services beneficial to the provision of a wide range of services that are capable of meeting the infinitely variable requirements of disparate secondary service providers and end users? In some ways, this is like asking if the "market economy" is a Good Thing.

We do not argue that services, 3rd party provision or a service marketplace are inherently bad. Instead, we suggest that the desire to attain the benefits that services so

enticingly offer and avoid, as far as possible, the risks associated with OTS service consumption is likely to lead to more and more standardization, and fewer, more dominant, providers, who will dictate the types of services offered, and the terms on which they are made available. Ultimately, the resulting hegemony will leave the majority of service consumers will little power to affect the services available.

We further suggest that this issue should be of concern to practitioners of SOC. Over-engineering solutions now on the basis that they may be required in the future is, in itself, not a solution. And it is unlikely that the answer lies in the type of "religious war" that has pervaded topics in technology in recent years. However, choices made now will affect how SOC develops and will determine if flexibility if considered as highly as cost. It just remains to be asked: will the majority of service consumers care?

Acknowledgements

References

[1] Carney, DJ, (1998), Quotations from Chairman David. Pittsburgh: Carnegie Mellon University, 1998.

[2] Clements, PC, (2001), "From Subroutines to Subsystems: Component-Based Software Development." In Councill, WT, Heineman, GT, (eds.): Component-Based Software Engineering: Putting the Pieces Together. Addison Wesley, 2001.

[3] Gregory, G, "Decision Analysis", Pitman Publishing, 1988.

[4] Help the Aged: "Post Offices": http://www.helptheaged.org.uk/en-gb/Campaigns/OtherIssues/PostOffice/ (15/6/06).

[5] Hofstadter, D, (1997), "Le Ton Beau de Marot", Basic Books.

[6] Szyperski, C.,(2001), "Components and Web Services", sdmagazine.com, August 2001: (A) Szyperski, C., "Components and Web Services", sdmagazine.com, August 2001: http://www.sdmagazine.com/documents/s=7208/sdm0108c/

[7] Times Online: "Big supermarkets face competition probe": http://business.timesonline.co.uk/article/0,,9074-2077540,00.html (9/3/06).

[8] ThisIsMoney: "Tesco's local rate mark-up": http://www.thisismoney.co.uk/money-savers/article.html?in_article_id=403818&in_page_id=5

[9] Brown, A., Johnston, S., Kelly, K.,(2002), "Using Service-Oriented Architecture and Component-Based Development to Build Web Service Applications", October 2002: http://www.rational.com/media/whitepapers/TP032.pdf.

[10] Wikipedia.org: IBM: http://en.wikipedia.org/wiki/IBM

[11] Wikipedia.org: Microsoft: http://en.wikipedia.org/wiki/Microsoft

[12] Wikipedia.org: Wal-Mart: http://en.wikipedia.org/wiki/Walmart