Towards a Killer App for the Semantic Web

Harith Alani, Yannis Kalfoglou, Kieron O'Hara, Nigel Shadbolt

Intelligence, Agents, Multimedia
School of Electronics and Computer Science
University of Southampton, UK
{ha, y.kalfoglou, kmo, nrs}@ecs.soton.ac.uk

Abstract. Killer apps are highly transformative technologies that create new markets and widespread patterns of behaviour. IT generally, and the Web in particular, has benefited from killer apps to create new networks of users and increase its value. The Semantic Web community on the other hand is still awaiting a killer app that proves the superiority of its technologies. There are certain features that distinguish killer apps from other ordinary applications. This paper examines those features in the context of the Semantic Web, in the hope that a better understanding of the characteristics of killer apps might encourage their consideration when developing Semantic Web applications.

1   Introduction

The Semantic Web (SW) is gaining momentum, as more researchers gravitate towards it, more of its technologies are being used, and as more standards emerge and are accepted. There are various visions of where the technology might go, what tasks it might help with, and how information should be structured and stored for maximum applicability [4][21][30]. What is certainly clear is that no-one who wishes seriously to address the problems of knowledge management in the twenty-first century can ignore the SW.

In many respects, the growth of the SW mirrors the growth of the World Wide Web (WWW) in its early stages, as the manifest advantages of its expressivity became clear to academic users. However, once the original phase of academically-led growth of the WWW was over, to the surprise of many commentators, the web began its exponential growth, and its integration with many aspects of ordinary life. Technologies emerged to enable users to, for example, transfer funds securely from a credit card to a vendor's account, download large files with real time video or audio, or find arbitrary websites on the basis of their content.

Most realistic visions of the SW include a version of this exponential growth. The SW infrastructure should be put in place to enable such growth. With a clean, scalable and unconstraining infrastructure, it should be possible for users to undertake all those tasks that seem to be required for the SW to follow the WWW into the stratosphere, such as publishing their RDF, converting legacy content, annotating, writing ontologies, etc.

However, that something is possible does not entail that it is inevitable. So the question arises of how developers and users might be persuaded to come to the SW. This type of growth of a network has often been observed in the business literature. Many technologies depend for their usability on a large number of fellow users; in this context

Metcalfe's Law [15] states that the utility of a network is proportional to the square of the number of users.

Technologies which have this effect are called *killer apps*. Exactly what is a killer app is to a large extent in the eye of the beholder; in the WWW context, killer apps might include the Mosaic browser, Amazon, Google, eBay or Hotmail. Hotmail attracted over 30 million members in less than three years after its 1996 launch; eBay went from nothing to generating 20% of all person-person package deliveries in the US in less than 2 years. Of course, the WWW was a useful enough technology to make its own way in the world, but without the killer apps it might not have broken out of the academic/nerdy ghetto. By extension, it is a hope of the SW community that the SW might take off on the back of a killer app of its own.

The dramatic development of the WWW brought with it a lot of interest from the business community, and the phenomenon of killer apps has come under much scrutiny [10][19]. Attempts have been made to observe the spread of killer apps, and to generalise from such observations; the tight development cycles of WWW technology have helped such observations.

In this paper, our aim is to consider the potential for development of the SW in the light of the killer app literature from the business community. Of course, it is impossible to forecast where the killer app for the SW will come from. But examination of the literature might provide some pointers as to what properties such an application might have, and what types of behaviour it might need to encourage.

2   Killer Apps and the Semantic Web

Killer apps emerge in the intersection between technology, society and business. They are technological in the broad sense of being artificial solutions to perceived problems, or artificial methods to exploit perceived opportunities (which is not to say that they need to have been developed specifically with such problems or opportunities in mind). Mere innovation is not enough. Indeed, a killer app need not be at the cutting edge of technological development at all. The killer app must meet a need, and be usable in some context, such as work or leisure or commerce. It must open up some kind of opportunity to bring together a critical mass of users.

To do this, killer apps have a number of features which have been catalogued by commentators. In this section, we will examine and reinterpret such features in the context of the SW. We reiterate that these features may not all be necessary, and they certainly are not sufficient; however they can act as an interesting framework to our thought on this topic.

The main point, of course, about a killer app is that it enables a superior level of service to be provided. And equally clearly, the SW provides an important opportunity to do this, as has been argued from the beginning [4]. There are obvious opportunities for any knowledge-based task or enterprise to improve its performance once knowledge sources are integrated and more intelligent information processing is automated.

2.1 The bottom line: Cost vs Benefit

However, merely providing the opportunity is not enough. Cost-benefit analysis is essential [10]. There are several aspects to costs. Obviously, there are financial costs; will people have to pay for the killer apps on the SW? Maybe not; there are many examples of totally free Internet applications, such as Web browsers, search engines, and chat messengers. Such applications often generate large revenues through online advertising. According to the Interactive Advertising Bureau UK[1] and PriceWaterhouseCoopers[2], the market size of online advertising in the UK for 2004 was £653.3m, growing more than 60% in one year. Free products may be very important in this context [19], and indeed killer apps are often cheaper than comparable alternative products [7].

But such costs are not the only ones incurred. There are also important resource issues raised by any plan to embrace the SW.

Conversion cost: As well as investing in technologies of certain kinds, organisations and people will have to convert much of their legacy data, and structure newly-acquired data, in particular ways. This immediately requires resources to support the development of ontologies, the formatting of data in RDF, the annotation of legacy data, etc., not to mention potential costs of exposing data in RDF to the wider world (particularly where market structures reward secrecy). Furthermore, the costs of developing smart formalisms that are representationally adequate (the fun bit) are dwarfed by the population of informational structures with sufficient knowledge of enough depth to provide utility in a real-world application [12]. Note also that such a process will require ascent of some very steep learning curves.

Maintenance cost: In a very dynamic domain, it may be that ontologies have to be updated rapidly [20][5]. The properties of ontologies are not as well-understood as they might be; areas such as mapping ontologies onto others, merging ontologies and updating ontologies are the focus of major research efforts. It is currently unknown as to how much such maintenance effort would cost over time.

Organisational restructuring costs: Information processing is integrated into an organisation in subtle ways, and organisations often subconsciously structure themselves around their information processing models [11], a fact implicitly accepted by the knowledge engineering community [27]. Surveys of organisations, for example, reveal that ontologies are used in relatively primitive ways; indeed, in the corporate context, the term 'ontology' is a generic, rarely defined catch-all term. Some are no more than strict hierarchies, some are more complex structures allowing loops and multiple instantiations, still others are in effect (sometimes multilingual) corporate vocabularies, while others are complex structures holding metadata [22]. Whatever their level of sophistication, corporate ontologies support the systematisation of large quantities of knowledge, far from the traditional AI view of their being highly detailed specifications of well-ordered domains. Ontologies may refer to an internal view of the organisation (marketing, R&D, human resources, etc) or an external one (types of supplier and supplies, product types, etc). A recent survey showed that only a relatively small number

---

[1] www.iabuk.net

[2] http://www.pwc.com/

(under a quarter) of corporate ontologies were derived from industry standards. The big issue for many firms is not representational adequacy but rather the mechanics of integration with existing systems [12].

Transaction costs: On the other hand, it is also true that if the SW does alter information gathering and processing costs, then the result will inevitably be some alteration of firms' management structures. The result will be leaner firms with fewer management layers, and possibly different ways of processing, storing and maintaining information. Such firms may provide opportunities for new SW technologies to explore, and a gap in the market from which a killer app may emerge.

It has long been argued that the size and structure of firms cannot be explained simply by the price mechanism in open competitive markets [8]. The allocation of resources is made using two mechanisms - first (between firms and consumers) by distributed markets and coordinated by price, but also (within firms) by the use of authority within a hierarchy (i.e. people get ordered to do things). The question then is how this relates to a firm's structure - when a firm needs some service, does it procure it from outside and pay a market price, or does it get it done in-house, using workers under some contractual obligation, and why?

It is generally thought that such organisational questions are determined by the *transaction costs* within a firm [33][34]. The promise of the SW is that many of the information gathering costs will be ameliorated. The general result of this is likely to be a continuation of trends that we have seen in economies since the widespread introduction of IT, which is the removal of middle management ("downsizing"), and the outsourcing of many functions to independent suppliers. In the SW context, of course, many of those independent suppliers could well be automatic agents, or providers of web services. If the SW contains enough information about a market, then we might well expect to see quite transformative conditions, and several market opportunities. The killer app for the business aspects of the SW may well be something that replaces the coordinative function of middle management.

But we should add a caveat here: the marginal costs of information gathering will be ameliorated, but equally there will, as noted above, be possibly hefty sunk costs up front, as firms buy or develop ontologies, convert legacy data to RDF, lose trade secrets as they publish material, etc. These initial costs may prove an extensive barrier to change.

Reducing costs: Here we see the importance of the increase in size of the user base. For example, the costs of developing and maintaining ontologies are high, but can be shared. Lightweight ontologies are likely to become more important [31]; not only are they cheaper to build and maintain, but they are more likely to be available off the shelf [22]. Furthermore, they are more likely to be easily understandable, mappable, maintainable, etc. The development of such lightweight multi-purpose ontologies will be promoted as the market for them gets bigger.

Similar points can be made about ontology development tools. Better tools to search for, build or adapt ontologies will spur their use or reuse, and again such tools will appear with the demand for them. And in such an environment, once an ontology has been developed the sunk costs can be offset by licensing the use of that ontology by other

organisations working in that domain. The costs, in such a networked environment, will come down over the period of use; if a single firm took on the costs of developing and licensing an ontology for a domain, that firm could also take on the maintenance costs. Organisations that specialised in ontology maintenance and training for users could spring up, given sufficient demand for their services.

Increasing Benefit: Similarly to data restructuring; there has to be some discernible benefit for organisations putting their data in RDF, and these benefits will become more apparent the more published data in RDF there is. So the issue here, which a killer app might help with, is that there seems to be little or no advantage for an individual firm in moving first. A firm that publishes its data in RDF early incurs costs early and takes a risk, but gets little benefit; and vice versa. Nevertheless, being first in a new market is a distinct advantage [10][7], but late entrants can also succeed if they outperform existing services [13] (e.g. Google). So there is a Prisoner's Dilemma to be sorted out.

Berners-Lee argues that the killer app for the SW is the *integration* [2]. Once distributed data sources are integrated, the sky becomes the limit. This of course could be true, but it will be hard to convince data providers to publish in RDF and join the SW movement without concrete examples of benefit. This is probably supported by Berners-Lee suggestion that we need to "Justify on short/medium term gain, not network effect" [3]. Integration alone might not be seen as a gain on its own, especially when considering costs and privacy issues.

In a survey for business use cases for the SW, researchers of the EU *KnowledgeWeb*[3] emphasised the importance of proper targeting for SW tasks [24] to avoid applying SW technologies to where they do not offer any clear benefit, which may discourage industry-wide adoption. The survey concluded that the areas which seem to benefit more from this sort of technology are *data integration* and *semantic search*. It was argued that these areas could be accommodated with technologies for *knowledge extraction*, *ontology mapping* and *ontology development*. Similarly, Uschold and Gruninger [31] argue that ontologies are useful for better *information access*, *knowledge reuse*, *semantic-search*, and *inter-operability*. They also list a number of assumptions to be made to progress towards a fully automated semantic integration of independent data sources. Fensel et al [14] describe the beneficial role of ontologies in general *knowledge management* and *eCommerce* applications. They also list a number of obstacles that need to be overcome to achieve those benefits, such as *scalable ontology mapping*, *instantiation*, and *version control*. Other obstacles, such as *trust*, *agent co-ordination*, *referential integrity*, and *robust reasoning* have also been discussed [18].

2.2 Leveraging Metcalfe's Law

The relevance of Metcalfe's Law, that the utility of a network is proportional to the square of the number of its members [15], is clear in the context of this examination of the nature of the costs It is often cited in other contexts as an explanatory variable for killer apps [10][19][13]. There are two stages to the process of growing a network; first get the network's growth accelerating, and second preserve the network once it is in place, in order to create a *community of practice (CoP)*.

---

[3] http://knowledgeweb.semanticweb.org/

Communities of practice: A CoP [32] is an informal, self selecting, group of people sharing some work- or leisure-related practice. The CoP that springs up around such a practice acts as a kind of support network for practitioners. It provides a language (or informal ontology) for people to communicate with, a corporate memory, and a means of spreading best practice.

This self-selection, and informality of CoPs, makes a community very hard to develop, because the community is a second-order development. So, we might take the example of Friend of a Friend (*FOAF*)[4]. FOAF is a basic ontology that allows a user to express simple personal information (email, address, name, etc) as well as information about people they know. Many SW enthusiasts considered FOAF to be *cool* and *fun* and started publishing their FOAF ontologies. Currently there are millions of FOAF RDF triples scattered over the Web, perhaps far more than any other type of SW annotations.

Social Network Applications: Surprisingly, there exist many Web applications that allow users to represent networks of friendships, such as Friendster[5], Okrut[6], LinkedIn[7], TheFacebook[8], SongBuddy[9], to name just a few. However, FOAF has simply become a more convenient form for representing, publishing, and sharing information. Even though none of the applications above are entirely based on FOAF, some have already begun reading and exporting FOAF files. FOAF is certainly helping spread RDF, albeit in a way limited to part of the SW community, and could therefore be regarded as a facilitator or a medium for possible killer apps that could make use of available FOAF files and provide some useful service.

Sustaining Network Growth: However, one interesting obstacle in the way of FOAF creating the nexus of users that will launch the SW is that a network is generally self-selecting and second order. One obvious benefit of FOAF is that, as a pretty simple ontology, it provides a relatively painless way of ascending the learning curve for non-users of SW technology. However, to sustain the network growth, there is still need to something underlying such networks, some practice, shared goal, or other practical purpose.

It may well be that a potentially more fruitful approach would be to support existing communities and try to expand SW use within them, so that little Semantic Webs emerge from them, as SW technologies and techniques reach saturation point within them [18]. And because CoP overlap, and converge on various boundary objects and other linking practices and artefacts [32]. There are many obvious aids to such a development strategy; for instance, good-quality ontologies could be hand-crafted for particular domains. But also, it turns out that a number of the best SW tools at the moment also support this "filling out" technique. For instance CS AKTive Space [28] specifically enables people to find out about the state of the discipline of computer science in the UK context; a limited but useful domain. Flink [23] generates FOAF networks

---

[4] http://www.foaf-project.org/

[5] http://www.friendster.com/

[6] http://www.orkut.com/

[7] https://www.linkedin.com/

[8] http://www.thefacebook.com/

[9] http://www.songbuddy.com/

for SW researchers. CS AKTive Space and Flink are winners of the 2003 and 2004 Semantic Web Challenges respectively.

Open systems and social aspects: One other useful aspect of Flink is that it integrates FOAF profiles with ordinary HTML pages, and therefore sets up an explicit link between the SW and the WWW. Direct interaction with other existing systems increases the value of a system by acquiring additional value from those systems [19]. One good example is Protégé, an ontology editor from Stanford [25]. By being open source and extendable, Protégé allowed many existing systems and tools to be linked or integrated with it, thus increasing its use and value. For this reason, and for being free, Protégé has quickly become one of the most popular ontology editing tools available today.

This openness is of course built into the very conception of the SW; the integration of large quantities of data, and the possibility of inference across them, is where much of the power stems from. As with the WWW, this does require a major programme of voluntary publication (e.g. to simply and conveniently compare prices across retailers over the WWW). The SW would add value (or reduce information processing costs) still more by allowing agents to do the same thing and more [17].

And as with the WWW, if this process takes off, then more and more vendors would have to publish their data in RDF, even if they are initially reluctant. The argument in favour of such coercion is that everyone benefits eventually, and that early movers not only gain, but force laggards to follow suit.

Privacy and Trust: Transparency and the removal of restrictions to publication are not undiluted goods. It may be that certain pieces of information benefit some organisations only as long as they withhold them from public view (trade secrets). Or that issues such as privacy and anonymity will rear their heads here. Or even that differing intellectual property regimes and practices will lead to competitive advantage being lost in some economies.

In particular, integrating large quantities of information across the Internet and reasoning across them raises potential problems. Firstly, it is the integration of information that threatens to allow harmful inference; information is quite often only harmful when seen in the right (or wrong) context. But the SW is the tool *par excellence* for doing that. And secondly, publication of information (e.g. FOAF) in a friendly and local context can quickly get out of one's control.

It is often argued that standard data protection legislation is adequate for the new online contexts, but that policing is the problem. As it stands, traditional restrictions on the gathering of information are becoming decreasingly relevant as information crosses borders so easily. More plausible is policing restrictions on how information can be used once collected.

Furthermore, formalising or externalising knowledge, for example in the creation of ontologies, can have a number of effects. First of all, knowledge that is codified can become more 'leaky', i.e. it is more likely to leave an organisation. Secondly, it will tend to reduce the competitive advantage, and therefore income, of certain experts. Thirdly, much depends on whether a consensus exists about the knowledge in the first place.

2.3 Creativity and risk

Killer app development cannot follow from careful planning alone. As we have noted already, there is no algorithm for creating a killer app. They tend to emerge from simple and inventive ideas; they get much of their transformative power by destroying hitherto reliable income streams for established firms. Christensen [7] points out that most killer apps are developed by small teams and start-ups. Examples include Google, eBay, and Amazon, which were all created by a few dedicated individuals. Giant industrial firms are normally reluctant to support risky projects, because they are generally the ones profiting from the very income streams that are at risk [10].

However, even though most semantic web applications have so far been built in research labs and small groups and companies, there is clear interest expressed by the big players as well. So, for example, Hewlett Packard has produced Jena [6], a Java library for building SW applications, and IBM has developed WebFountain [16], a heavy platform for large scale analysis of textual web documents using natural language analysis and taxonomies for annotations. Adobe has perhaps gone further than many; Acrobat v5 now allows users to embed RDF metadata within their PDF documents and to add annotations from within Web browsers which can be stored and shared via document servers.

The SW provides a context for killer app development, a context based on the ability to integrate information from a wide variety of sources and interrogate it. This creates a number of aspects for the potential for killer apps. First of all, SW technologies might essentially be expected to enable the retrieval of data in a more efficient way that possible with the current WWW which is often seen as a large chaotic library.

On the other hand, it may be that the SW might take off in an original and unpredictable direction. The clean infrastructure that the W3C ensures is in place could act as a platform for imaginative methods of collating and sifting through the giant quantities of information that is becoming available. This might result in a move away from the webpage paradigm, away from the distinction between content providers and consumers, as for example with efforts like CS AKTive Space [28], or a move towards a giant, relatively uniform knowledge base (of the CYC variety) that could cope with all those complexities of context that foiled traditional AI approaches [1]. The ultimate vision of the SW that prevails should affect not only the standards developed for it [21], but also where we might look for killer apps.

2.4 Personalisation

Personalisation has been a common thread in the development of killer apps. Customers tend to become more loyal to services they can customise to their liking [10]. Many of today's killer apps have some level of personalisation; Amazon for example makes recommendations based on what the customer buys or looks at; Auto Trader[10] and Rightmove[11] save customers' searches and notify them via emails when a new result to their query is available; personalised web services attract more customers (if done properly!) and provide better tailored services [13].

---

[10] http://www.autotrader.co.uk
[11] http://www.rightmove.co.uk

Personalisation is often the key to providing the higher service quality than the opposition. The service itself need not be provided in any better ways, but the personalised aspect gives it the extra that is needed to defeat the alternatives. Such a connection could be indirect; for example, an Amazon-style recommender system, linked with an advertising platform, could help find alternative revenue streams and therefore drive down the cost to the consumer.

It goes without saying that personalisation is a hot topic on the SW, as well-annotated knowledge sources can be matched against RDF statements about individual consumers, to create recommendations or targeted products. There may well be major advantages to be had in systems that can feed information discreetly into recommender systems [26][9].

Nevertheless, it is the personalisation aspect that has much potential for the SW, as long as the provision of enough information for the system to work interestingly is not too painful.

2.5 Semantic Web Applications

There have been few sustained attempts to try to promote SW applications. For instance, the important work of the W3C naturally is focused on the standards that will create the clean platform that is a necessary but sadly not sufficient condition for the SW to take off. But one of the most interesting and inspired is the series of Semantic Web challenges, which we will discuss briefly in the next section.

3 The Semantic Web Challenge

The annual International Semantic Web Challenge[12] (SWC), has been a deserved success, sparking interest and not a little excitement. It has also served to focus the community. Applications should "illustrate the possibilities of the Semantic Web. The applications should integrate, combine, and deduce information from various sources to assist users in performing specific tasks." Of course, to the extent that it does focus the community, the SWC will naturally influence the development of the SW.

Submissions to the SWC have to meet a number of minimum requirements, viz:

- First, the information sources used
  - should be geographically distributed,
  - should have diverse ownerships (i.e. there is no control of evolution),
  - should be heterogeneous (syntactically, structurally, and semantically), and
  - should contain real world data, i.e. are more than toy examples.
- Second, it is required that all applications assume an open world, i.e. assume that the information is never complete.
- Finally, the applications should use some formal description of the meaning of the data.

Secondly, there are desiderata that act as tiebreakers.

---

[12] http://challenge.semanticweb.org/, where the criteria quoted below are to be found.

- The application uses data sources for other purposes or in another way than originally intended
- Using the contents of multi-media documents
- Accessibility in multiple languages
- Accessibility via devices other than the PC
- Other applications than pure information retrieval
- Combination of static and dynamic knowledge (e.g. combination of static ontologies and dynamic work-flows)
- The results should be as accurate as possible (e.g. use a ranking of results according to validity)
- The application should be scalable (in terms of the amount of data used and in terms of distributed components working together)

In the light of our discussion above, these are interesting criteria. Many of them are straightforwardly aimed at ensuring that the characteristic possibilities of the SW are realised in the applications. For instance, the SW would have little point indeed if it only worked on toy examples, did not scale, could not work with distributed information sources, or if it required some kind of closed world to work.

However that may be, what is of interest here is the relation to the SWC criteria and the literature on killer apps. What the SWC is intended to uncover are new ways of exploiting information, particularly distributed information, and demonstrating the power of interrogation. In this respect, the challenge can only raise the profile of the SW, and help extend its community to more people and organisations. The SWC is an excellent vehicle for demonstrating where added value may come from. And as we have seen, increasing the size of the network will bring with it exponentially-increasing benefits.

But the SWC looks unlikely to furnish us with a (prototype of a) killer app, because the criteria focus on interesting results, rather than on usability, superiority or the alteration of old habits. Some of the criteria are slightly double-edged. For example, it is essential that an application uses a formal description of the meaning of the data. This, of course, is a deliberate attempt to ensure that one of the most contentious aspects of the SW (one of the most commonly-cited causes of scepticism about the SW's prospects) is incorporated. That is, the use of ontologies to provide understanding of terms in knowledge from heterogeneous sources. However, the way the challenge is constructed means that what is bound to happen in many if not most applications is that the developers will create their system with a possibly very painstakingly constructed ontology in mind, rather than taking the more difficult option of employing a very lightweight system that could work with arbitrary ontologies. The situation is somewhat similar to the knowledge engineering Sisyphus challenges, where KE methodologies were tested and compared by being applied to the same problem. However, as the methodologies were applied by their developers, the results were less than enlightening; a later attempt to try to measure how difficult methodologies were to use by non-specialists suffered from an unwillingness of most developers to discover this key fact about their methods [29].

There is little here to create the genuine community (as opposed to a large network); to promote the idea that users have something of a responsibility not to free

ride, and to publish RDF data. Neither is there much to promote personalisation within that community. There is little to protect privacy, little to reduce the pain of annotating legacy data or building ontologies, and, although the focus of the SWC is the results of information-processing, little to ensure that such processing can integrate into the organisational workflow. Surprisingly few of the traditional requirements, from a business perspective, appear in the SWC criteria.

None of this, let us hastily add, is intended as a criticism of the SWC, which has publicised the SW and drawn a lot of attention to the extra power that it can provide. Our point is merely that there is a lot more to finding a killer app than producing an application that does brilliant things.

4 Discussion

Killer apps must provide a higher service quality, and evolve (pretty quickly) into something perceived as indispensable, conferring benefits on their users without extra costs or steep learning curves. Individual users should coalesce into a community of practice, and their old habits should change in accordance with the new possibilities provided by the app. This is particularly important as the SW is likely to impose new costs on users in the short term, for example through having to annotate legacy content, develop ontologies, etc.

As the SW is in a relatively early stage of development, it is not currently clear exactly what threats and opportunities it provides (and, of course, the future form of the SW will conversely depend on what applications for it are successful). There has been some speculation about how the SW will develop, and what extensions of the WWW will be appropriate or desirable. For instance, consider a recent attempt by Marshall and Shipman to understand potential development routes for the SW [21], which sets out three distinct but related visions of the SW.

I *SW technology could bring an order and consistent structure to the chaotic Web. So information access would be assisted by semantic metadata.* This vision envisages that humans will continue to be the chief agents on the SW, but that information could now be represented and stored in ways to allow its use in situations far beyond those foreseen by its original authors. In other words, the SW will extend the existing Web, but exactly how is hard to predict. In order for potential SW Killer Apps to respond to this vision should, ideally, have the following properties in particular:

- They should help foster communities of users (that is, at some level, users should want to interact, and share experiences, with others). SW technology is expected to facilitate knowledge sharing and bringing more people together.
- Users should not feel submerged in a mass, but should retain their individuality with personalised products. With more machine readable information becoming available (eg FOAF), better personalisation should be feasible.
- Users should be able to bring as much of their legacy content up to date with relatively painless maintenance techniques. So, for example, applications should be able to leverage comparatively simple ontologies; ontology construction, merging and selection should be made easier, and we should be able to move

away from handcrafting; annotation methods and interfaces have to be easy. In all these cases, the existence of a community of interested users will provide the initial impetus for the user to ascend the learning curve.

II *The Web will be turned, in effect, into a globally distributed knowledge base, allowing software agents to collect and reason with information and assist people with common tasks, such as planning holidays or organising diaries.* This vision seems close to Berners-Lee et al's SW grand vision [4]. In many ways it is the composition of the other visions, assuming machine processing and global representation of knowledge. It is also a vision that will require more from a potential Killer App:

- They should exploit integrated information systems to make inferences that could not be made before. Showing added value is key to encourage businesses and content providers to participate in the SW.
- They should help remove the rather painful need to annotate, build ontologies, etc.
- The new application should fit relatively smoothly into current work or leisure experiences. Little change in habits is acceptable, assuming some returned benefit, but too much change is a problem!

III *The SW will be an infrastructure, made up of representation languages, communication protocols, access controls and authentication services, for coordinated sharing of knowledge across particular domain-oriented applications.* Information is used largely for the original purposes of its author, but that much more machine processing will take place. If this vision prevails, then a prospective SW Killer app should pay special attention to the following:

- It should not compromise other important aspects of users' lives, for instance by threatening privacy to a dangerous degree, either by making inappropriate surveillance possible, or by facilitating torts such as identity theft.
- Furthermore, if such a vision comes to pass, then the opportunities for killers apps are all the greater, in that any such standards-driven platform approach should make it possible for as many applications to flourish on top of it as possible. Whether such applications will ever be acknowledged as SW apps rather than general WWW applications is open to debate!

These conditions for each vision of the SW are, of course, necessary yet not sufficient! Furthermore, all of the conditions apply, to some degree, to each of the three visions.

We have seen that killer apps appear when there are opportunities to make progress on costs, communities, creativity and personalisation. All new technologies begin with a handicap in these areas. They impose costs, of retooling, of learning curves and of business process rescheduling. There is always a chicken and egg problem with the development of a community of users – the technology of necessity precedes the community. The risks of creative thought become clearer at the outset; the benefits only appear later. And the dialectic between personalisation and creating economies of scale often means that the latter are pursued long before the former. As an added handicap, it is often the case that the costs are borne disproportionately by early adopters.

The opportunities of the SW are also therefore counterbalanced by the risks. We note that we cannot predict where new killers will come from. The transformations

that such applications wreak make the future very different from the present. Hence we can't be concretely prescriptive. But the general requirements for killer apps that emerge from our review of the business/management literature suggest certain routes for development, in addition to sensible lists of characteristics such as the criteria for the SW challenges, or the conditions listed at the beginning of this section.

So it is probably uncontroversial to assume that any SW killers will have to provide (1) a service that is not possible or practical under more traditional technologies, (2) some clear benefit to developers, data providers, and end users with minimum extra costs, and (3) an application that becomes indispensable to a user-base much wider than the SW researchers community. But additionally, research should be focusing on four important areas. First of all, perhaps most important, the cost issue should be addressed. Either the potentially large costs of annotating, ontology development, etc, should be mitigated, or side-stepped by thinking of types of application that can work with minimal non-automatic annotation, low cognitive overhead, or ontologies that sacrifice expressivity for simplicity. Secondly, another way of improving the cost/benefit ratio is to increase benefits, in which case the fostering of user communities looks like a sensible way forward. This means that applications in real-world domains (preferably in areas where the Internet was already important, such as media/leisure or e-science) look more beneficial than generic approaches. Thirdly, creativity is important, so radical business models are more interesting than simply redoing what the WWW already does. And fourthly, personalisation needs to be addressed, which means that extended user models are required.

When we look at the three visions outlined by Marshall and Shipman, vision III appears to be the one most amenable to the development of killer apps, in that it envisages a platform upon which applications sit – the form of such applications is left relatively open. In contrast, vision I, for example, doesn't see too much of a change for the WWW and the way it is used, and so there are fewer opportunities opening up as a result. Indeed, when we look at vision I, assuming that the SW does improve the navigation of the chaotic web, it might even be appropriate to say, not that there is a killer app for the SW, but rather that the SW is the killer app for the WWW. Whereas, with Marshall and Shipman's vision III, the vision is of a garden in which a thousand flowers bloom. On this vision, it is the painstaking, pioneering and often tedious negotiations of standards that will be key; such standards need to support the right kind of research.

5 Conclusions

Killer apps are very difficult things to monitor. They are hard to describe, yet you know one when you see one. If you are finding difficulty persuading someone that something is a killer app, it probably is not! A lot depends on the bootstrapping problem for the SW – if the SW community is small then the chances of someone coming up with a use of SW technology that creates a genuinely new use for or way of producing information are correspondingly small. For it is finding the novelty that is half the battle. There is unlikely to be much mileage in simply reproducing the ability to do something that is already possible without the SW. Furthermore, it is likely that a killer app for the SW will exploit SW technology integrally; merely using RDF will not quite do the trick

[30]. The willingness of the producers of already-existing killers to use SW technology, like Adobe, is encouraging, but again will not necessarily provide the killer app for the SW. The checklist of criteria for the SWC gives a good list of the essentials for a genuinely SW application.

This is not simply a matter of terminology. The SW is more than likely to thrive in certain restricted domains where information processing is important and expensive. But the ambitions of its pioneers, rightly, go beyond that. For that to happen, killer apps need to happen. We hope we have given some indication, via our examination of the business literature, of where we should be looking.

Acknowledgments

References

1. T. Berners-Lee. The semantic web road map. *http://www.w3.org/DesignIssues/Semantic.html*, 1998.
2. T. Berners-Lee. Iswc2003 keynote. http://www.w3.org/2003/Talks/1023-iswc-tbl/, 2003. 2nd International Semantic Web Conference (ISWC2003), Florida, USA.
3. T. Berners-Lee. Www2004 keynote. http://www.w3.org/2004/Talks/0519-tbl-keynote/, 2004. 13th Int. World Wide Web Conf., New York.
4. T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.
5. S. Buckingham-Shum. Contentious, dynamic, information-sparse domains. and ontologies? *IEEE Intelligent Systems*, pages 80–81, Jan/Feb 2004.
6. J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. Jena: Implementing the semantic web recommendations. Technical report, HP Laboratories Bristol, HPL-2003-146, Dec. 24, 2003.
7. C. M. Christensen. *The Innovator's Dilemma*. Harvard Business School Press, 1997.
8. R. Coase. *The Nature of the Firm, Economica 1937, reprinted in Oliver E. Williamson & Sidney G. Winter (eds.) The Nature of the Firm: Origins, Evolution and Development*. Oxford: Oxford University Press, 1991.
9. S. Cox, H. Alani, H. Glaser, and S. Harris. The semantic web as a semantic soup. In *Proc. 1st Workshop on Friend of a Friend*, Galway, Ireland, 2004.
10. L. Downes and C. Mui. *Unleashing the Killer App*. MIT Press, Harvard Business School Press, 2000.
11. K. Eischen. The social impact of informational production: Software development as an informational practice. Cgirs working paper 2002-1, Center for Global International and Regional Studies, University of California, Santa Cruz, 2002.

12. J. Ellman. Corporate ontologies as information interfaces. *IEEE Intelligent Systems*, pages 79–80, Jan/Feb 2004.

13. P. Evans and T. Wurster. *Blown to Bits*. Harvard Business School Press, 2000.

14. D. Fensel, C. Bussler, Y. Ding, V. Kartseva, M. Klein, M. Korotkiy, B. Omelayenko, and R. Siebes. Semantic web application areas. In *Proc. 7th Int. Workshop on Applications of Natural Language to Information Systems (NLDB 2002)*, Stockholm, Sweden, 2002.

15. G. Gilder. Metcalfe's law and legacy. *Forbes ASAP*, 13 September 1993.

16. D. Gruhl, L. Chavet, D. Gibson, J. Meyer, P. Pattanayak, A. Tomkins, and J. Zien. How to build a webfountain: An architecture for very large-scale text analytics. *IBM Systems Journal*, 43(1):64–76, 2004.

17. J. Hendler. Agents and the semantic web. *IEEE Intelligent Systems*, pages 30–37, March/April 2001.

18. Y. Kalfoglou, H. Alani, M. Schorlemmeret, and C. Walton. On the emergent semantic web and overlooked issues. In *Proc. 3rd Int. Semantic Web Conf. (ISWC)*, Japan, 2004.

19. K. Kelly. *New Rules for the New Economy: 10 Radical Strategies for a Connected World*. Penguin Books, 1998.

20. M. Klein and D. Fensel. Ontology versioning on the semantic web. In *Proc. 1st Int. Semantic Web Working Symp.*, pages 75–91, Stanford University, CA, USA, 2001.

21. C. Marshall and F. M. Shipman. Which semantic web? In *Proc. 14th HyperText Conf. (HT'03)*, pages 57–66, Nottingham,UK, 2003. ACM.

22. D. L. McGuinness. *Ontologies Come of Age*. in D. Fensel, J. Hendler, H. Lieberman, and W. Wahlster editors. Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential, MIT Press, 2002.

23. P. Mika. Social networks and the semantic web: The next challenge. *IEEE Intelligent Systems*, 19:82–82, 2005.

24. L. Nixon. Prototypical business use cases. FP6 IST NoE Deliverable D1.1.2, KnowledgeWeb EU NoE - FP6 507482, Dec. 2004.

25. N. F. Noy, M. Sintek, S. Decker, M. Crubezy, R. W. Fergerson, and M. A. Musen. Creating semantic web contents with protege-2000. *IEEE Intelligent Systems*, pages 60–71, March/April 2001.

26. M. C. schraefel, A. Preece, N. Gibbins, S. Harris, and I. Millard. Ghosts in the semantic web machine? In *Proc. 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web*, Galway, Ireland, 2004.

27. G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. V. de Velde, and B. Wielinga. *Knowledge Engineering and Management: The CommonKADS Approach*. MIT Press, 1999.

28. N. Shadbolt, m. schraefel, N. Gibbins, and S. Harris. Cs aktive space: or how we stopped worrying and learned to love the semantic web. In *Proc. 2nd Int. SW Conf*, Florida, 2003.

29. N. Shadbolt, K. O'Hara, and L. Crow. The experimental evaluation of knowledge acquisition techniques and methods: history, problems and new directions. *International Journal of Human-Computer Studies*, 51:729–755, 1999.

30. M. Uschold. Where are the semantics in the semantic web? *AI Magazine*, 24(3), 2003.

31. M. Uschold and M. Gruninger. Ontologies and semantics for seamless connectivity. *SIGMOD Record*, 33(4), 2004.

32. E. Wenger. *Communities of Practice: Learning, Meaning and Identity*. Cambridge University Press, 1998.

33. O. E. Williamson. *Markets and Hierarchies*. New York: Free Press, 1975.

34. O. E. Williamson. *The Nature of the Firm: Origins, Evolution and Development*. Oxford: Oxford University Press, 1991.