Workshop on Engineering e-Business Systems

Report on the Panel: Future Trends

David Akehurst and *Rogério de Lemos*
Computing Laboratory
University of Kent at Canterbury

The intent of the Panel was to discuss the future of e-Business systems and how these types of systems should be engineered.

The Panel was chaired by Alan Cameron Wills (Trireme International Ltd.), and had the participation of Keith Bennett (University of Durham), José Fiadeiro (ATX Software and University of Lisbon), Stuart Kent (University of Kent at Canterbury), and Wolfgang Emmerich (University College London).

The panel started by each of the panellist giving initial statements. Keith Bennett started by enumerating the essential difference between e-Business software and conventional software: e-Business software is driven by the emergent organisations, which are associated with continually evolving business. The distinct feature of these systems is that of maintenance, which is not simply fixing bugs but primarily adding and testing new functionality. The vision that should be assumed is how we make software evolve by a factor of 100%. It is envisaged that the solution should be at the architectural level, which should include the integration of legacy systems. The high factor of evolution can be achieved by regarding software as a service and not a product, where the software is used rather than owned.

José Fiadeiro started by identifying 'agility' as a main feature of e-business systems because these systems have to survive an ever-changing world. There is still no sound methodology for coping with the complexity of modelling e-business and the systems used to support it. Model driven approaches should be able to relate different models and facilitate the integration between functional and non-functional requirements. There is likely to be a move towards 'service oriented development', where instead of software providers offering products, the development will move towards a customer driven request for particular 'software service'.

Stuart Kent mentioned that there is a need to customise processes because of distinct features of future software. This is currently achieved at the lowest technological level since it gives the necessary flexibility. However, it requires specialist craftsmen and there are not enough to go around, for example, security is a high skilled job. A move to higher levels of abstraction might facilitate less specialised engineers being able to perform the changes, leaving the specialists to provide the mappings from high to low levels. The challenge of model driven approaches is to make this possible without loosing the necessary flexibility, moreover, model driven approaches should be applied to the development of languages and tools.

Wolfgang Emmerich said in his statement that essentially e-business systems are software intensive systems that support a business. Current business models are very restrictive, and software usually mirrors the business model. It is well known that the introduction of technology without changing the business, barely leads to an increase of productivity. There is an opportunity for business methods to change in order to take full advantage of the new technologies. In addition to business-to-business transactions, there is also the opportunity for peer-to-peer (or customer-to-customer) e-business. The new business should not be constrained by the business models being used for the last hundred years. There is the need for new models of business that allow different types of deployment for the different business solutions, in which new technologies have to be considered, for example, trusted third parties.

After the initial statements, workshop participants put various questions to the Panel, in order to provoke discussion surrounding the future trends of e-Business.

Taking as a basis the IBM slogan for the future of e-business, which states that "a million enterprises having a billion people using a trillion devices", Steve Cook asked whether the Panel thought the slogan was realistic? If so how will it change the way we design/build/run systems? Keith Bennett answered that software has been supplier driven, and because of the pervasiveness of computers this is about to change to demand driven. José Fiadeiro said that the issue is the complexity (and the interaction) and not the numbers, and the problem was how to control/predict the behaviour of all interactions. Stuart Kent said that these devices would know a person in a thousand different ways, and the challenge is to integrate their services considering there are many legacy systems. Steve Cook made a comment that these devices would be Linux servers, which fit into a microchip and cannot be seen by the naked eye. In Wolfgang Emmerich's opinion, it should be very easy to achieve the thousand devices that anyone can own. These would be wearable devices in touch with the environment to know what services the environment might be able to provide. Morris Sloman added that the functionality of these devices should be very simple because there are so many devices to design. They will need to be self-configurable, as there are too many for each one to be separately configured, hence there may be a trend towards the mechanisms employed by biological systems. Keith Bennett said that is too difficult to design complex systems, and that might lead to self-adaptation based on emerging properties and behaviour. Wolfgang Emmerich made the remark that solutions like the RosettaNet tend to constraint flexibility because of the rules that are imposed.

Leonor Barroca raised the issue that as computing devices become smaller and electronic, and electronic relationships become more transient and opportunistic, how trust can be established between devices? The Panel answered that branding will have a big effect on trust, which currently does not really play a part. Also there will be the need to balance publicity of personal data against the level and suitability of services offered or received. For example, in publicly accessible diaries, who can read/modify/make its entries? Social issues definitely will play a part.

Regarding model driven software development Anneke Kleppe asked a question whether models grow to be the "programming" language of the next decade? She continued by saying that, nowadays nobody reads or cares about the assembler code generated by your compiler. In the future, nobody will care about the Java code generated from a model.

Stuart Kent disagreed and replied that while compilers generate everything, modelling has to know what is underneath. Wolfgang Emmerich said that it was the feeling when compilers were being introduced. The problem today is that UML has very specific domain assumptions, thus there is the need to encode variables in very high-level language. Stuart Kent added that the intention of raising the language might lead to better business models, but not executable programs. He continued by saying that so far the abstraction level has not been raised above object-orientation. Instead of moving to general-purpose languages, we are moving into domain specific languages. Perhaps in the future the assembler of business-to-business will be XML.

At the end of the Panel, Alan Cameron Wills asked to the panellists what would be the role of academics in the field. Keith Bennett answered that academia should be funded long-term research; nothing to do with the research that industry is doing now. It should analyse the problems that industry is having now, abstract from these, and look for solutions that can be used in general systems. José Fiadeiro mentioned there is a need for blue-sky research at the academic level. Stuart Kent said that software engineering is related to the engineering of large systems. On one hand, academia needs to have an understanding of these types of systems; on the other hand, industry has no time to reflect on the problems. Hence the people from industry should have to time to reflect while academics should dirty their hands. Wolfgang Emmerich made the remark that the problem is so big that cannot be solved by PhD's or Post-Docs alone, the problem has to be tackle at a different scale. The dilemma is that academic training is currently biased towards training individuals, while industry wants teams.