For Official Use **DSTI/IND(2008)28**

Organisation de Coopération et de Développement Économiques
Organisation for Economic Co-operation and Development **10-Oct-2008**

English - Or. English

DIRECTORATE FOR SCIENCE, TECHNOLOGY AND INDUSTRY
COMMITTEE ON INDUSTRY, INNOVATION AND ENTREPRENEURSHIP

DSTI/IND(2008)28
For Official Use

CHAPTER 3. USER PERSPECTIVES ON SOFTWARE FUNCTIONALITIES

OECD PROJECT ON INNOVATION IN THE SOFTWARE SECTOR

This document is submitted for information and discussion under Item 6 a) of the CIIE agenda. It does not incorporate feedback received from the meeting of the Software Advisory Expert Group that took place in Tokyo on 7 October 2008.

Contact: Douglas Lippoldt, E-mail: douglas.lippoldt@oecd.org Tel. (33-1) 45 24 19 26
Piotr Stryszowski E-mail: Piotr.Stryszowski@oecd.org Tel. (33-1) 45 24 91 30 and
Jeoung-Yeol Yu E-mail: Jeoung-Yeol.Yu@oecd.org Tel. (33-1) 45 24 98 74

JT03252716

English - Or. English

CHAPTER 3
USER PERSPECTIVES ON SOFTWARE FUNCTIONALITIES

Introduction and main points

1. As software becomes more deeply embedded in social and economic infrastructure, the range of functionalities sought by users is broadening and the exigencies for performance with respect to the various functionalities are increasing. In some cases, software innovators are creating new markets (as happened in recent years with on-line music sales), while in other cases the demand is user-led with software developers responding with new or improved products (as happened with security enhancements in home computing systems). New or enhanced functionalities are sometimes introduced by the original software developers, but also by follow-on developers writing applications or even modifications of the original product (*e.g.* as sometimes happens with open source systems).

2. Both original and follow-on developers may bring new functionalities for existing products to the market, incrementally or individually. In some cases, developers and platform owners also move to bundle in expanded sets of functionalities – including via acquisition of those developed by add on innovators. These may be made available via periodic major new releases of the software products. This bundling in of existing functionalities, in turn, may spur new rounds of innovation in functionality by developers aiming to generate renewed traction in markets.[1]

3. The various functionalities of a software package together shape the user decision to employ the product; they also shape the user experience. Generally, multiple functionalities are sought from individual products. In many cases, the specific functionalities are not an end but rather a means to obtain a desired aspect of performance from a software product. For example, a user may seek reliability in a software package, but ultimately have the objective of using the software for a more concrete purpose. For example, a software package designed to monitor a critical piece of safety equipment may also need to meet very high standards in terms of reliability.

4. This chapter addresses user perspectives and functionalities by considering an illustrative set of functionalities. This set highlights examples that have become priorities for many users as well as the manner in which the software sector has innovated to deliver them. These include security and privacy, mobility, interoperability, accessibility and reliability. An annex to the chapter considers how software is transforming industries beyond the traditional computer industry, in part by adding new functionalities to software in their products.

5. Main points from the analysis include:

- As economy-wide activities are deeply affected by software, issues relating to software functionalities have come to concern stakeholders across society. Software functionality can have systemic effects with respect to the economy (*e.g.* enabling new forms of organisation) and society more broadly (*e.g.* social inclusion).

[1] *E.g.* see Parker and Van Alstyne (2008).

- Market demand for software functionality plays an important role in propelling technological innovation by providing signals and incentives for innovators to act. At the same time, the nature of software as a digital, non-rivalrous product means that there can be large returns to scale for innovators that are able to respond to this demand.

- Due to the technologically heterogeneous and complex nature of software functionality, there is an increasing emphasis on collaborative and, in some cases, open innovation approaches to development of improved functionality. The diversity in the content and in the technologies means that individual firms or developers face challenges in delivering comprehensive solutions and generally must draw on resources beyond the walls of the firm in order to assemble the necessary elements for success.

- The contribution of user knowledge and experience to the development of software functionality has emerged as an important source of input for innovation processes; the software sector is a leading sector in the engagement of users, a factor that is contributing to the dynamism of the innovation activity.

- The amount of software in many modern products, from automotive to consumer electronics, is growing at rapid pace, and for many industries software has become the heart of their new products. Software has become a differentiator and enabler overall increases in product performance and new functionality. Moreover, software developers are reaping the benefit of large economies of scale and low marginal costs by marketing their products, adapted, to many other sectors including reuse of embedded software.

Software Functionalities and User Perspectives

6. This Chapter considers five selected functionalities in order to illustrate the types of innovation underway in software, taking into account user perspectives. The functionalities covered include security and privacy, mobility, interoperability, accessibility and reliability.[2] The Chapter reviews the definition of these functionalities and examines issues surrounding their evolution including user participation (Box 3). An Annex to the chapter underscores the role of businesses beyond the software sector as users and developers of software functionalities.

Security and privacy[3]

Background

7. Over the last 20 years, Information and Communications Technologies (ICTs) have become essential for governments, businesses and individuals in most economic and social activities. Information technologies and the use of the Internet in particular provide a powerful driver for innovation, growth and social well-being. As a result, economies and societies increasingly depend on ICTs. For example, which key activity, public or private, could be carried out today if its information technology component were not available? Which business sector could continue to operate, which government branch could continue to

2 The choice of functionalities for examination in this section was based on the priority interests of the OECD Member Countries as specified in the scoping document for the project [OECD document code: DSTI/IND(2007)3, paragraph 11].

3 This section draws on a paper prepared by Nick Mansfield, consultant to the OECD, under the supervision of the Secretariat for the OECD Information, Computer and Communications Policy Committee (ICCP), Working Party on Information Security and Privacy (WPISP), and in consultation with the Secretariat for the Committee on Industry, Innovation and Entrepreneurship (CIIE).

provide services if suddenly the Internet were to collapse? Even Critical Infrastructures (CI, *e.g.* water, energy, transport) rely increasingly on the effective functioning of ICTs. Information security and privacy protection are two important and inter-related policy areas in relation to ICTs.

8. With regards to information security, our societies' reliance on ICTs and the interdependence of users due to the generalisation of Internet usage was recognised by OECD member countries in the 2002 *Guidelines for the Security of Information Systems and Networks* ("Security Guidelines"; OECD, 2002). These Guidelines aim to promote a culture of security among all participants as a means for protecting information systems and networks, from government infrastructures to global corporate systems and home personal computers. The Guidelines provide a set of high level policy and operational principles that create a general frame of reference to help participants (whether government, business and individual user) *i)* understand security issues; *ii)* respect ethical values in the development and implementation of coherent policies, practices, measures and procedures for the security of information systems and networks; and *iii)* take responsibility according to their role. Since the adoption of the Guidelines, governments and businesses have developed policies and initiatives to foster a culture of security for information systems and networks and have implemented a number of measures to support them (OECD 2005a and 2004a; BIAC/ICC 2003 and 2004).

9. Ensuring ethical use of ICTs and respect for fundamental values is another essential objective. When organisations use software to process data relating to identified or identifiable individuals (personal data), individual's privacy and liberty can be impacted. The 1980 OECD *Guidelines on the Protection of Privacy* ("Privacy Guidelines"; OECD, 1980) represent a long-lasting consensus on a set of fundamental principles to enable the economic and social potential of information technologies to be realised while protecting privacy. These Guidelines are reflected in numerous national and regional legislative and regulatory instruments, as well as self-regulatory privacy codes of conducts, practices and policies, in both OECD and non-OECD countries. Effective implementation of these privacy frameworks is however challenged as ever more information systems are connected to the Internet, data processing routinely takes place anywhere in the world, personal data flows across borders, is stored, mirrored and processed in more distributed ways than ever before.

10. Both the OECD Security and the Privacy Guidelines reinforce each other: the Privacy Guidelines include a Security Safeguards principle that the Security Guidelines contribute to addressing.[4] The Security Guidelines include principles that aim to ensure that security measures support and remain compatible with essential values such as privacy.[5] Both are considered essential by OECD countries, as evidenced by the draft Ministerial declaration for the Seoul Ministerial Meeting on the Future of the Internet Economy (17-18 June 2008) and its supporting report on Shaping Policies for the Future of the Internet Economy (OECD, 2008g).

Software security and privacy

11. Software is one of the main components of ICTs. Software determines what the system does and how it is done, which is essential with regards to both security and privacy protection: whether and how data, including personal data, can be collected and stored, how it is processed, whether it can be linked to

4 "Personal data should be protected by reasonable security safeguards against such risks as loss or unauthorized access, destruction, use, modification or disclosure of data."

5 "Ethics: Participants should respect the legitimate interests of others […] Democracy: the security of information systems and networks should be compatible with essential values of a democratic society. Security should be implemented in a manner consistent with the values recognized by democratic societies including […] the confidentiality of information and communication, the appropriate protection of personal information, openness and transparency."

other data, shared and with who, how long it will remain in the system. All these aspects and many others are determined by software functionalities. Software privacy may not be sufficient to ensure full privacy protection, but it can play a significant role to provide more efficient privacy protection

12. Software security is also essential as a large number of threats seek to exploit software vulnerabilities or vulnerabilities resulting from software deployment and configuration. For example, one of the main threats to the Internet Economy and to critical information infrastructures is the development of malicious code called Malware, including viruses, worms and trojans. Malware has evolved from occasional "exploits" in the mid-1990s to a global multi-million dollar criminal industry. It is a form of software that is inserted in information systems exploiting software vulnerabilities and that could be used to launch cyber attacks for money extortion, information theft (*e.g.* identity theft), espionage or – hypothetically – terrorism (OECD, 2008c.).

13. "Privacy" and "security" in the context of software can be interpreted as meaning software with embedded privacy and security features (*e.g.* access control features like password protection), as features that are coded in a secure or privacy friendly way (*e.g.* via secure coding or privacy by design approaches) or as *specialised* functional software (*e.g.* a firewall or antivirus software package). As specialised privacy and security software is a relatively small market compared to the global software market as a whole, and as the scope of the project encompasses the software sector as a whole, this paper focuses mainly on embedded privacy and security features in software rather than on the smaller specialised security and privacy software market.

Risk and risk tolerance

14. Risk can be defined as a level of probability that negative consequences or impact on an activity could occur. Any activity carries with it a certain level of risk. Security and privacy are a means to manage some of this risk with the objective of maintaining it at an acceptable level.[6] They are implemented by adopting measures (also called "controls") to prevent and mitigate risk. Nevertheless, whatever the measures taken, risk can never be completely eliminated. Absolute security and privacy do not exist. For any activity, there is always a remaining level of risk that must be accepted. Otherwise, as is often quoted, "*the only truly secure system is one that is powered off, cast in a block of concrete and sealed in a lead-lined room with armed guards - and even then I have my doubts*" (Eugene H. Spafford[7]).

15. An important driver of demand for security and privacy in software is the acquisition of a greater degree of control over perceived and real risk related to the use of the software. There is always a requirement to balance costs versus safety, security, privacy, and other functionalities (*e.g.* performance, accessibility, connectivity, scalability, interoperability) and the remaining risks must be acceptable to the user. However, there is a security paradox[8]: the user may become less tolerant of remaining risks, exactly because things have become safer. The more preventative measures are taken, the less tolerance there may be for remaining risks and unforeseen disasters. Yet, beyond a certain point, costs may become prohibitively expensive relative to the benefits in terms of increased safety and security.

16. Risk may be seen as the probability of a given event associated with exposure to negative impacts, in this context resulting from exploitation of vulnerabilities or weaknesses. When both the probability and the impact are low, users are usually unconcerned, feel secure and accept the risk. When

---

6 See principle 6 « Risk assessment » of the OECD Guidelines for the Security of Information Systems and Networks.

7 Professor of Computer Science, Purdue University.

8 See OECD (2008a), Section 2, "Government authorities and agencies".

both the probability and the impact are high enough, users become concerned and can be motivated to take measures to reduce or control the risk.

17. In business terms, the level of acceptable risk is known as the "risk tolerance". The overall risk tolerance varies according to an infinite list of parameters ranging from the specific context, purposes and expected benefits from the use of the software to intangible and unquantifiable factors such as societal, cultural and individual values and beliefs that vary across the world. The tolerance for tangible risk elements such as online fraud can be quantified as direct measurements can be made, but it is not the case for many other components of a given user's risk tolerance, such as the disclosure of personal information. At a macro level, only very general trends can be identified.

Information security and privacy

18. Information security traditionally relates to the confidentiality, integrity and availability of information. In software development, confidentiality is most commonly applied to access controls, the protection of stored data and data communications; integrity is most commonly applied to software processes and the data being processed; availability is often expressed in terms of the software processes and information being available when and where required, often blending both software processes and telecommunications delivery mechanisms. In some instances, availability is related to interoperability between programmes sharing and exchanging data. Closely linked to these three dimensions are audit and appraisal processes that can lead to some level of assurance. The level of assurance is a measure of the comfort the security and privacy controls provide. Audit can give some comfort that the controls are working properly. Appraisal can give comfort that the security is adequate. The 2002 OECD *Guidelines for the Security of Information Systems and Networks: Towards a Culture of Security* (OECD Security Guidelines) provides a set of nine principles that apply to all participants (*i.e.* businesses, governments, individual users) at all levels to promote a culture of security as a means of protecting information systems and networks.[9]

19. Privacy has several interpretations based on cultural and legal understandings. In its simplest form, it can be expressed as an internationally recognised fundamental value: respect for and protection of privacy, including the protection of personal data. Privacy can also be expressed as the ability of an individual or groups to seclude themselves or information about themselves and thereby reveal themselves selectively (privacy in this case is often referred to as "the right to be left alone"). In software terms, privacy protection is most often linked to the control of personal data. Personal data is defined as information relating to identified or identifiable individuals. Control includes the limitation of collection, use, storage, disclosure to third party, linkage with other data, etc. The 1980 OECD *Guidelines on the Protection of Privacy and Transborder Flows of Personal Data* (OECD Privacy Guidelines) set out core principles to assist government, business and consumer representatives in their efforts to protect personal data. They represent an international consensus on general guidance concerning the collection and management of personal information.

20. Ultimately, from a software innovator's perspective, software security is about identifying, implementing and operating the technical measures that will control and mitigate risk to reduce it to an acceptable level to the user. While, from the same perspective, the same logic can be applied to privacy protection, in reality, software privacy is often about translating into technology high level principles such as the one set forth in the OECD Privacy Guidelines or in national laws. Technology choices within a company usually follow on from the corporate operating security and privacy policies that, in turn, are derived from the high level business principles followed by the organisation.

---

9 BIAC/ICC (2003, 2004).

21. From a business perspective, the benefits of security spending are sometimes difficult to measure because they relate to uncertainties (risks). However, they can be more easily connected to the overall objectives of the user or his organisation than the benefits of investments related to privacy protection. The latter are much more difficult to rationalize from the sole perspective of the interest of the data controller[10]. This difference may explain why the frameworks for privacy protection are often provided by law or government policy.[11]

Security and privacy controls

22. To reduce the level of risk, there are three general types of security and privacy controls: technical, administrative and physical. Their objectives may be to prevent, detect or correct risks. Technical controls correspond to technology measures. In a software context, they include the security and privacy features of the software itself, *e.g.* password access control features or audit log capacity. Administrative controls concern the manner in which the system is managed through policies, procedures and standards in relation to the operation of the system (*e.g.* password management policies, backup recovery procedures or analysis of audit logs). They also correspond to actions performed by users of the software (*e.g.* users choosing and remembering strong passwords and keeping them secret), or system administrators analysing audit logs generated by the software. Physical controls rely on the application of risk reducing physical barriers and deterrents such as alternate power sources or data backup devices. A holistic approach that uses all three types of controls has to be applied to reduce risk as security is only as strong as its weakest part.

23. Users commonly perceive a software as secure or privacy protective when it includes a number of security and privacy features. However, from a more comprehensive security and privacy perspective, the level of risk is appropriately reduced only when the security and privacy features included in the software are *operated* within a broader security *management* strategy that reflects the acceptable level of risk for the user. This implies that the user:

- Has an accurate understanding of the risk in terms of threats and vulnerabilities, and their potential negative consequences and impact on their activities,
- Knows the level of risk that is considered as acceptable,
- Applies the technical measures that can prevent and mitigate risk to reduce it to that level, and
- Implements the operational and the management measures that can provide the appropriate context for the technical measures to be efficient.

24. Software security and privacy are sometimes seen to relate primarily to the technical dimension of security and privacy. This is an important dimension, but the simple existence of software security controls does not as such reduce the level of risk effectively unless these technical measures are operated and managed according to good practices. For example, a given software application may include a password feature but if users choose weak passwords such as their name or birthday, or if they keep their long passwords handy on a sticker under their keyboard, it is unlikely that risk be reduced to an acceptable level. Similarly, audit log features that keep track of all the actions performed by the software and users of

---

10 The data controller is the person who « is competent to decide about the contents and use of personal data » (OECD Privacy Guidelines). It is often the operator of the software that processes personal data.

11 The difference in the nature of the principles set forth in the OECD Security Guidelines and Privacy Guidelines reflect this fundamental difference between security and privacy.

the software can be embedded in the software, however the audit logs generated have to be analysed by a skilled administrator to detect unusual patterns, errors or possible attacks.

25. In general, ensuring adequate software security entails costs for system operators in all three control elements: technical, administrative and physical controls. Investment in education of users may be costly, but also has the potential to yield some significant reduction in these expenses.

Functional Requirements

26. A key element for users is the assurance that the software code is fit for the purpose for which it was intended. At a high level, the goal to protect security and privacy seems simple, though, in practical terms, it requires a precise understanding of the risk context, or risk assessment, which often involves resources and planning. Determining how "fit" is measured is even more difficult because it has to "fit" the business scenario and perceptions of the risk tolerance of the user. Users will balance a wide range of critical or important functional requirements, such as mobility, scalability, interoperability, security, privacy, or ease of audit, just to name a few. In specific circumstances, such as banking, these can be better defined and software developed and tested to fit requirements. In general scenarios, the looser the "fit" then the less effective software becomes and the more difficult to link it to operational environment. On the other side, there is a danger that if security and privacy are too tight, the software will become expensive or restrictive.

27. Audit processes ensure that the controls in place are operating as they are intended and appraisal processes ensure the controls are fit for the purpose intended. Where software is used in a defined environment, such as in a business, audit and appraisal can be considered as a possible administrative option. But in the consumer market where the consumer tends to invest only the minimum in security and privacy management and operation, they are not practical. That said, much has been done to ensure that these functions are made more available and in a more useable manner for general users. Internet service providers (ISPs), for example, have sought to embed more security protections into their underlying service offerings, which allows the ISP to provide the audit and appraisal functions on behalf of its customers.

Role of security and privacy features in users' software choice

28. Security and privacy software features certainly affect the user's choice, but to an extent that is hard to measure. It is difficult to distinguish between the user being motivated to select one piece of software rather than another for the embedded security and privacy features as opposed to other features. The primary reason for acquiring software is usually not its security features but its capacity to perform tasks that meet user's needs. The extent to which users are ready to dedicate money and resources for security and privacy to perform these key tasks depends on a large number of variables, ranging from level of comfort with technology to purpose of purchase. Security and privacy is only a subset of the user's overall software cost and benefit of ownership, a subset that can include both a cost for the acquisition of software and for its management, operation and support. Furthermore, as noted above, security and privacy management costs are usually much higher than software acquisition cost for the user. Ultimately, there is no simple methodology to measure, within the overall software cost of ownership, how much users consider the investment for software security and privacy – among other features – to be worth.

Trends

29. Several major trends related to the evolution of ICTs and ICT usage can be identified that might influence the role played by security and privacy in software:

- ICTs have become ubiquitous and critical in most people's lives
- ICT connectivity and mobility are going global
- Software is shifting from a product to a service, in some areas
- Crime has migrated on-line
- Software is growing increasingly complex
- Users are exhibiting reduced tolerance for risk.

Supply of security and privacy functionality

30. On the supply side, a number of processes and practices exist for developers and innovators to enhance security and privacy in software, such as code evaluation practices, security and privacy by design methodologies, better coding practices as well as standards and other security and privacy assurance mechanisms. Some of these processes and practices are actually increasingly used by software developers (*e.g.* code evaluation, better coding practices), others are limited to niches where security assurance plays a particular role (*e.g.* standards in the defence sector). The factors enabling or impeding their use are varied and complex.

31. Risk is a function of threats exploiting vulnerabilities and, while there can be vulnerabilities at all levels including human (*e.g.* poor passwords), procedural (*e.g.* poor training) and organizational (*e.g.* poor oversight), a major source of vulnerability is related to the quality of the software code. While it could be extreme to consider that security and privacy in software is "all about coding"[12], it is recognized that good coding practices and code evaluation are essential elements for security and privacy. What is important, however, is that an efficient process for code evaluation and vulnerabilities identification and resolution is in place.

32. Government policies can be an important driver to protect the security of information systems and networks and the privacy of individuals although it is unclear how much they foster *software innovation* for security and privacy. Procurement processes for government and private sector acquisition can also play a role in improving privacy and security. A better understanding of the underlying incentive structure for security and privacy in software could certainly help foster software innovation in this area. Tailored policies to support innovation in the area of software security and privacy could create positive externalities and help develop a culture of security and privacy and, more broadly, foster trust and confidence in the Internet economy.

User Expectations

33. Understanding individual privacy and security needs is challenging. Large organisations can affect resources for this exercise, formulate their requirements in relatively formal terms and assess the efficiency of the measures in place through audit and appraisal. It may be that smaller organisations and home users sometimes fall back on the basic expectation to have the same level or better security and privacy in a digital environment as enjoyed in their physical one. In business terms, this equates to having a similar risk appetite in both environments. They may have a tendency to expect technology to deliver what they need without them having to define it and they often have a heightened awareness and sensitivity to threats and vulnerabilities without knowing what to do.

12 Dewar, R. and Schonberg, E. (2008).

34. Software innovation takes place at the functional and operational levels on the supply side of markets as a consequence of user demand. Fortunately, companies often place many savvy technical and consumer-oriented employees in key roles to ensure that end users do not have to express their concerns or needs directly to developers, or vice-versa. Companies which focus on cyber-security and usability for end-users will benefit from meeting those marketplace needs.

35. As the software paradigm shift takes hold and some forms of software increasingly move from being a product to being a service, all participants will need continuous education in order to safely exploit the opportunities emerging, including transferring individual skills on personal risk management in the physical environment to the digital one. This aspect of security and privacy in software is about changing individual behaviours and creating cultures of security and privacy, as well as technological responses by software developers to needs articulated through market demand. Creating a culture of security and privacy that changes behaviours regarding the use of information technologies is also about users adopting a habit of "learning for life".

36. Reducing this gap requires raising awareness and education of all participants. Individual behaviour might also be motivated to change by appropriate policies and technology measures (*e.g.* security and privacy usability). Changing individual behaviour to adopt a culture of security and privacy is, in any case, a long term challenge.

Main Elements and Summing Up

37. Software can include safeguards and controls to protect privacy as well as prevent and mitigate security risks. Security and privacy are important dimensions for software users, playing an essential role for economies and societies that are increasingly reliant on ICTs, including for their critical infrastructures, day to day business, government operations and individual activities. Software innovation in these areas can *directly* benefit software market players at a micro level and can also *indirectly* improve the overall level of trust and confidence in the Internet economy and society and the economy as a whole.

- Users' demand for security and privacy in software depends on the level of risk that they are ready to tolerate ("risk tolerance"). Users' risk tolerance varies according to a large number of context-specific factors, including societal, cultural and individual values.

- Security and privacy features embedded in software play an important role for risks reduction but they need to be operated and managed according to good practices and within an overall risk management strategy.

- Security and privacy software features affect users' choice but to an extent that is hard to measure. Often, privacy and security are not the primary reasons for selecting a software package and the relevant features may not be easily distinguished from other secondary software features in some products. With disparate investments into processes (*e.g.* audit) and planning leading to a precise understanding of their risk context, in some instances it can be difficult for users to assess properly whether the embedded security and privacy features in software are "fit for purpose", leaving them exposed to unacceptable privacy or security risks.

- A few general trends seem to indicate a likely decrease of users' tolerance to risk related to software: *i)* the increased dependence of all users on ICTs for their essential activities, *ii)* the shift from software as a product to "software as a service"[13], *iii)* the migration of crime online (*e.g.*

[13] User perceptions of security issues in this regard are sometimes reported in the media, *e.g.*, see articles from *Byte and Switch* (07.04.08) http://www.byteandswitch.com/document.asp?doc_id=150418 and

fraud via malware), *iv)* increased user awareness for security and privacy following government policies such as data breach notification legislation.

- Quantifying security and privacy in relation to software is challenging: privacy and security metrics are tied to many difficult to quantify or variable parameters.

Elements on processes for software security and privacy

- Several layers of an organisation are involved in the development and integration of security and privacy into software: *i)* the overall objectives of the organisation are expressed at the business level, *ii)* the operational level defines why the functionality is required, *iii)* the functional level defines what functionality should be provided and *iv)* the software development level determines how the functionality is provided. In some cases, software innovation takes place at that bottom level, far removed from the user or business level, with little precise knowledge of the users' risk tolerance, giving rise to greater risks. Other business models integrate security and privacy more thoughtfully into the objectives of the business, which translates into improved development, operation and functionality.

- Secure and privacy-friendly software can be developed using waterfall or iterative software development models, each having their strengths and weaknesses. "Security and privacy by design" approaches can also be adopted, as encouraged by the OECD 2002 *Security Guidelines* and recent OECD work on privacy. In the area of privacy, there are some mechanisms to help software developers check their products against privacy protection criteria to assess the effectiveness of their design and protect privacy from the outset.[14] However, so far these standards are not globally adopted and utilized. Furthermore, though there is a certain degree of consistency at a high conceptual level, privacy related regulations vary across countries creating challenges to the implementation of privacy requirements at the practical software engineering level.

- Code development and evaluation processes, whether in a closed or open environment, play a key role to eliminate vulnerabilities and should not be confused with the software licensing regime.

- Privacy and security assurance mechanisms might be enhanced by following international guidance, good and best practices, as well as formal and informal standards. Self declarations, certificates, trustmarks and seals might also be used to achieve this objective. While privacy requirements are often expressed in general, legal and technology neutral terms, there are no definitive software/technical processes, methodologies or mechanisms to translate these concepts into functional technical requirements that can be implemented by software developers. However, there are now privacy seals available for software, available though some initiatives recently launched in this area. In the area of information security, a number of security assurance mechanisms exist to provide assurance regarding the integrity of the software (*e.g.* code signing) or assurance that the security can be demonstrated to be fit for purpose, through international and national standards, reputation of the vendor, vendor and third party certification schemes.

TechNewsWorld (27.02.07) http://www.technewsworld.com/story/55971.html (both articles last accessed on 29.09.2008).

14 See, for example, Microsoft's *Privacy Guidelines for Developing Software Products and Services*, available at: http://www.microsoft.com/downloads/details.aspx?FamilyID=c48cf80f-6e87-48f5-83ec-a18d1ad2fc1f&displaylang=en (last accessed on 29.09.2008).

Mobility

Introduction

38. Electronic delivery of information and access to digital content are becoming ubiquitous, driven by the technological advancement of broadband wireless networks, platforms and software. Network convergence and widespread diffusion of high-speed broadband have created new opportunities for innovation in software. Platforms are increasingly mobile with respect to points of access to the fixed infrastructure and are evolving with some independence from specific systems. The emphasis on mobility is reshaping the ties between the software components and the network hosts where they execute, resulting in software that retains a higher degree of bandwidth utilization, flexibility and robustness than in the past.

39. These developments have opened up space for new market demand, business opportunities, productivity and innovation. Across the economy, users are demanding computing structure that is mobile and available at any time and any location.

Types of software mobility

40. Mobility is an important quality for products in the current environment where information processing has been thoroughly integrated into everyday objects and activities. Experience over the last few years with mobile data and information communications has led to increased demand for mobile applications and services for the general public as well as business, especially in those business processes that involve workers on the move. Developments in middleware are supporting the improved integration of wide ranges of mobile networks and computing devices.

41. There are different kinds of mobility schemes, such as terminal mobility, personal mobility, and service mobility. From the software perspective, mobility in a ubiquitous system includes physical mobility and connectability of devices and non-physical modes such as the ability to function in diverse environments.[15]

Main drivers for mobility

Enhanced broadband and wireless networks

42. One of the key factors enabling mobility is the increased capability of broadband wireless networks. Until recently, mobile networks did not have enough capacity or sufficient bandwidth to guarantee a good user experience with respect to sophisticated digital content. Now, with heavy investment in broadband infrastructures, especially among OECD member economies, high-speed digital networks are becoming a reality. In particular, implementation of 3G mobile telephony is well underway and the number of subscribers to 3G services is expanding (Table 1).[16]

---

15 Niemela and Latvakoski (2004) offer a more technical presentation of three types of mobility: 1) Actual mobility - An extension in the capability of an autonomous software agent that enables the dynamic transfer of code and data towards nodes containing relevant resources. Exploitation of actual agent mobility can save network bandwidth and increase reliability and efficiency of the execution; 2) Virtual mobility - The ability to recognise and function in an environment with multiple, networked options for execution; and 3) Physical mobility – The possibility for mobile and wireless computing devices to connect to the Internet from dynamically changing access points.

16 3G refers to the third generation of standards for mobile phones developed under the auspices of the International Telecommunication Union.

Table 1. Current 3G subscribers

System	Number of countries	Number of subscribers
cdma2000	72 (mainly America, Asia)	275.2 million
W-CDMA	55 (mainly West Europe and Japan)	70 million
HSDPA (3.5G)	36	N.A.

Source: OECD (2006b).

43. The current 3G deployment provides greater bandwidth and increased functionality for delivery of content in a mobile fashion. Already as of the first half of the present decade, 3G networks were delivering access speeds ranging from 128 Kbps to nearly 2 Mbps (OECD 2004). With speeds greater than 200 kbps, users are able to rapidly perform such tasks as downloading music albums, conducting video conferences and playing interactive games. Enhanced broadband capabilities facilitate convergence of data, video, Internet and multimedia services, and these services are now available over a broader geographical area.

44. Other wireless technologies are also being developed at rapid pace, facilitating connection of a broad range of electronic devices, including over wide and local areas. Examples of particular relevance in this context include WiFi (wireless fidelity), WiMAX (Worldwide Interoperability for Microwave Access), iBurst and WiBro. For example, WiMax can deliver last-mile wireless broadband access without the need for direct line-of-sight to a base station. Although it is in the beginning stages of commercialization, some services are already in place. In Korea during 2006, telecom companies launched "WiBro" services around the Seoul area, offering connectivity even when the user is moving, but the take up has been slow.[17]

Technical advancement of mobile platforms

45. Mobile device manufacturers are working with content providers, software industries and other industry participants to develop handsets and features that facilitate access to and use of mobile contents. Mobile devices such as smartphones, PDAs (personal digital assistants), GPS electronics (global positioning system) and handheld games have gained computing power and functionality. For example, mobile phones today provide multiple arrays of functions such as voice, data, music, photographs and games. Smartphones, devices that while providing voice telephony, also run an operating system allowing developers to code to the machine level and control every facet of the device, are proliferating. A typical smartphone or PDA has a processing capability equivalent to personal computer back in 1990s. IDC, a provider of ICT market intelligence, forecasts that shipments of converged mobile devices (*e.g.* smartphones and PDAs) will grow from 124.6 million in 2007 to 376.2 million in 2012.[18]

46. Several different operating systems on mobile devices have been developed with a variety of client-side execution environments such as Windows CE, Symbian and Palm OS. Shipments of PDA devices loaded with mobility-friendly operating systems are tending to rise (Table 2). In the case of mobile game devices, programs such as Java 2 Micro Edition (J2ME) and BREW have been developed in order to optimise for use in small devices. Because of the small screens in mobile devices, software firms are

17 Other wireless technologies operate in a personal area for very local mobility. These include RFID, NFC and Bluetooth technologies, among others.

18 IDC (2008), *Worldwide Converged Mobile Device 2008–2012 Forecast and Analysis: A Category Comes of Age*, http://idc.com/getdoc.jsp?containerId=211431.

developing special software (*e.g.* Opera Mobile, Scope, jig browser) that allows users to look at the Internet more easily and to provide full browsing function.

Table 2. Worldwide Preliminary PDA Vendor Shipment Estimates by Operating Systems, 1Q2007 (Units)

	1Q07 (shipment)	1Q07 Market Share (%)	1Q06 (shipment)	1Q06 Market Share (%)	1Q06-1Q07 Growth (%)
Windows CE	3,184,703	62.1	1,937,667	52.8	64.4
Research In Motion	928,239	18.1	929,883	25.3	-0.2
Palm OS	314,353	6.1	489,220	13.3	-35.7
Symbian	288,000	5.6	132,000	3.6	118.2
Linux	33,400	0.7	43,530	1.2	-23.3
Others	377,150	7.4	137,000	3.7	175.3
Total	**5,125,845**	**100.0**	**3,669,300**	**100.0**	**39.7**

Note: Excludes smartphones, such as Treo 750 and BlackBerry 81xx, but includes cellular PDAs such as BlackBerry 87xx.
Source: Gartner Dataquest (May 2007).

Emerging mobile market

47. As software-driven technology becomes more mobile, new markets are emerging to take advantage of new opportunities. The growing demand for mobile content – such as music, games, data and multimedia materials – is providing further incentives for innovation by software developers as well as by stakeholders in the telecommunications, electronics and media industries. In addition to rather conventional services such as ringtones, news, and games, higher levels of services such as financial services and mobile television are being brought to the market. As the market develops, numerous actors – including users – are taking part in various parts of a complex and changing value chain. Yankee Group estimate the mobile data service to reach USD 146 billion in 2009 and Portio Research predict mobile content market to be more than triple to USD 59 billion over the four years to 2009 (OECD 2006b).

Mobility in enterprise and business

48. The rapid growth of wireless networks, mobile functionality and mobile devices are having impacts on business well beyond the ICT sector. Some of early economic analysis shows that mobility can have major effects on labour productivity at firm level (OECD, 2004). Consequently, it is not surprising that enterprises are putting increased priority on "mobilizing" their workforce in order to make them more effective and available.

49. According to an IDC forecast, the number of mobile workers in the US will increase from 105 million in 2006 to over 120 million by the end of 2011 and for Europe, the number is expected to grow from 84 million to 91 million over the same period. IDC estimates that the mobile enterprise application market will grow from USD 1.2 billion in 2005 to 3.5 billion in 2010, representing a compounded annual growth rate of 23%.[19]

50. As businesses look at an increasingly mobile workforce, they are seeking ways to permit secure access to corporate information over mobile devices. Device-based security (such as "device wipe" or locking and encryption of data) is among the most important features that a mobile security solution must

[19] Information from the IDC web site, downloaded in June 2008 from the following location, http://www.idc.com/getdoc.jsp?containerId=prUS20491506.

offer. Besides maintaining secure access, the mobile enterprise applications deliver contents beyond email; the ability to view documents, participate in video conference and access company software tools. Applications such as multi-party conference calling, dedicated voice key, push-to-talk over cellular and internet call (VoIP over WLAN) are being used to reach mobile workers.

51. The expansion of "M-commerce" has generated growing demand for mobile device security software from both businesses and consumers. According to a forecast from IDC, worldwide mobile security license and maintenance revenue exceeds $200 million in 2007 and will continue to grow at a healthy annual growth rate through the forecast period. In the future, features like mobile firewall, mobile VPNs (virtual private networks), and mobile antivirus protection are expected to gain increased importance for people seeking to secure use of expanded mobile functionality IDC (2007),

Demand and implication for software sector

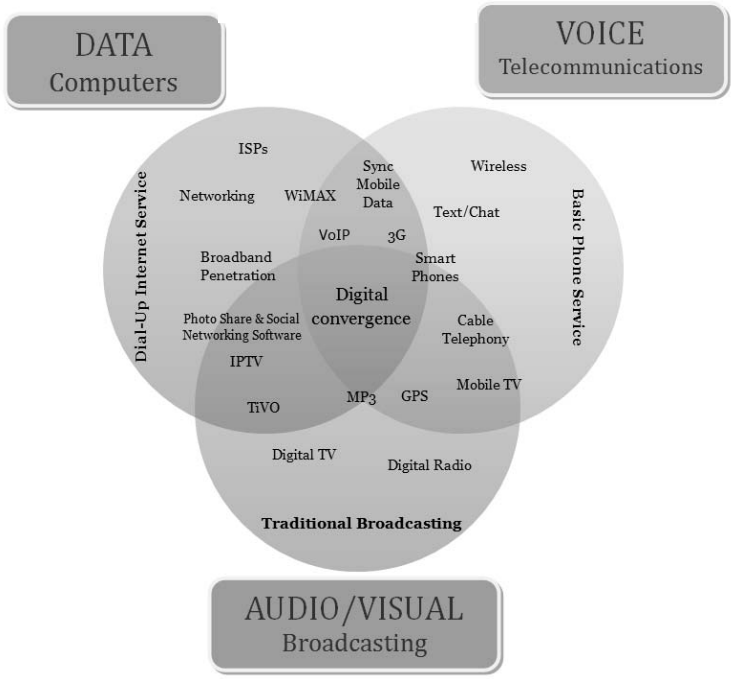
52. The increasingly diverse and geographically disbursed computing environment poses a number of opportunities as well as challenges for software developers. One of the challenges is finding ways to support mobility (Sousa, 2002). Market demand is encouraging developers to find new ways to enable users to take advantage of local capabilities and resources in a given environment, while also providing access to external users, devices and resources that operate in change conditions. This means that the software cannot be designed to rely on a fixed set of hardware devices; it must adapt itself to available devices. Often, the response is to aim for a software architecture that reduces the amount of device-specific development to the maximum extent possible. Software-as-a-service and cloud computing (discussed in Chapter 1) offer platform-independent mobile access, for example.

53. The same logic can apply to a user who may change context (*e.g.* when travelling with a laptop and PDA), with consequences for communication and system structure. This means that although the set of hardware devices is known, the structure of the communication software components must be able to adapt to itself to the user's context, which requires support from the software architecture (Williams, 2004). Applications should be able to follow the user and move seamlessly between devices. Also, since mobile software interacts with software located at other network platforms, mobile software must be robust in the face of intermittent connectivity.

Summing up

54. Mobility has become a key software functionality and it is in demand across the various software markets. This characteristic is now featured across a broad range of business and consumer applications. The utility of this functionality has been widely recognised and its inclusion in software has become de rigor in many situations. Together with enhanced hardware capabilities, this realisation is promoting innovation for better integration of systems regardless of the geographical position. It is also related to technological convergence, whereby the development of mobility as a software functionality is associated with connectivity and multifunctionality in devices with data, voice and audio-visual applications (Figure 1).

Figure 1. Examples of Convergence in Data, Voice and Audio-Visual Technologies



Source: US Department of Commerce.

Interoperability[20]

55. The term interoperability is used in various contexts and with various meaning. According to one report of the International Organisation for Standardization, interoperability may be defined as "the capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the users to have little or no knowledge of the unique characteristics of those units".[21] The EU Software Directive 1992 defines interoperability between computing components generally to mean "the ability to exchange information and mutually to use the information which has been exchanged". The U.S. E-Government Act of 2002 defines interoperability as "the ability of different operating and software systems, applications, and services to communicate and exchange data in an accurate, effective, and consistent manner".

56. The key to interoperability can be found in the ability of outsiders to have access to the structure of the technical interfaces of software to which a connection is desired. The Internet is among the most obvious vehicles for interoperability, whereby any software can connect and exchange data as long as it adheres to the key protocols.

---

20 The concept of reliability is applied here with respect to software primarily in an operational sense. There are other dimensions such as programmatic interoperability, which describes the manner in which organisations work together to achieve interoperability. More information on these aspects can be found, for example, on the web site of the Software Engineering Institute at Carnegie Mellon University, here: http://www.sei.cmu.edu/ (last accessed 08.08.2008).

21 ISO (2003), Proposed Draft Technical Report for: Information technology -- Learning, education, and training -- Management and delivery -- Specification and use of extensions and profiles, International Organisation for Standardization, available at: http://old.jtc1sc36.org/doc/36N0646.pdf.

57. In considering the different aspects of interoperability, one might distinguish between two dimensions: vertical and horizontal interoperability (Weiler, 2005). Vertical interoperability can be seen as access between applications providers and platform owners. Application program interfaces (APIs) enable interaction between their software products and depend on availability of interface information to application providers for a given platform. Achieving vertical interoperability is a complex task that is a result of co-operation between software developers. This is because vertically related products are highly integrated and may be provided by many developers. For example, access to the Internet involves several layers of vertically related products and applications; hardware, operating software, and application platforms such as a Web browser.

58. Horizontal interoperability can be seen as developer access between different operating systems. Horizontal interoperability may be established by porting the APIs or relevant interfaces to various operating systems or platforms through co-operation among developers who recognize the mutual needs and benefits. For example, many components of the Internet communication standards such as TCP/IP, the World Wide Web and domain names, have been developed through consensus process and are now accepted by the marketplace.

59. Even though there are competing definitions of interoperability, their main goal is rather clear: to enable heterogeneous software products and services that are components of information and communication technology systems to work harmoniously. This can promote an increase in user confidence, value and choice, as well as competition among providers, by removing technical impediments to the use of products and services from various vendors. With interoperable products and services, the user may not need to choose a specific technology or change equipment as often. Depending on the context, interoperability may also help foster the acceptance, success and penetration of new technologies.

The demand for interoperability

60. The software industry is undergoing an evolution where systems are built by assembling heterogeneous, off-the-shelf or proprietary software and by integrating these with external systems. As this evolution continues, developers and some users are learning new techniques and processes for building systems in this new mixed environment; this requires a deep knowledge of integration and interoperability issues. An increase in technical heterogeneity drives a greater demand for data and information integration as stakeholders try to find better ways to innovate and improve their ICT performance.

61. From the supplier's perspective, the heterogeneity leads to increased market demand for their solutions to be capable of successfully working in a mixed ICT environment. Businesses, for example, are looking for any opportunity to reduce costs. Developers no longer enjoy the luxury of developing every component nor using an array of adapters for every unique requirement. Instead, applications that are built using interoperable techniques speed up development and reduce cost. Interoperability increases firm revenue and efficiency of business process by allowing system synergies to be exploited and more cost effective technologies to be utilized.

62. The ability to deliver a product to interoperate with other products or services can have an impact on market demand for a product. In many cases, market forces and competition have provided incentives for developers to make products that are interoperable. Developers that fail to adhere to widely-adopted industry standards or fail to provide a minimum degree of access to their products may expose themselves to risk of losing competitive edge over those firms whose products do implement such standards or provide access for external developers. This risk can potentially be compensated in cases where the developer is able to bring into the market an alternative product of significantly higher quality (*e.g.* in cases where the developer introduces a breakthrough innovation).

63. Interoperability may also have economic consequences such as network externalities. If competitors' products are not interoperable due to causes such as lack of coordination, then there can be an exacerbation of market consequences (either increased market share or failure). At the same time, the substitutability achieved by interoperability can imply larger market size which may result in lower unit costs and prospects for better prices for consumers.

The need for interoperability in other sectors

64. Public sector and business activities rely heavily on information technologies. An important aspect of software in ICT intensive sectors (*e.g.* in aerospace, mobile telephony, petroleum, pharmaceutical and automotive) lies in its ever increasing complexity and fast rate of technology evolution. Individual suppliers are increasingly finding that collaboration is a critical dimension in meeting the demands driven by ever evolving business environment. It is virtually impossible for even the biggest and leading companies to supply the sophisticated functions needed by all potential users of its software. This leads to much emphasis in achieving software and data interoperability as the reach of particular software products extends from industry to industry and across various applications. Thus, interoperability can have a variety of implications for a modern economy.

65. There can be a variety of benefits from establishing interoperability Braunschweig (2005) provides an illustrative list:

- easy, cost-effective and error-free data transfer between applications,
- easy expansion of functionality in application software,
- avoidance of vendor lock-in,
- support to quality assurance process, and
- trigger for innovation.

At the same time, promotion of interoperability may also entail potential costs such as monetary costs to developers, possible security risks or undue constraints on innovation.

66. Government also influences market demand for interoperability, both as a customer and a policymaker. As customers, governments increasingly rely on ICT systems to carry out their respective missions. In procurement of systems and services for various operations, governments often consider interoperability as one selection criterion.[22] In part this can be driven by operational concerns. From a policy perspective, governments sometimes address interoperability concerns from the perspective of promotion of innovation or competition.

67. For example EU policy initiatives have brought interoperability to centre-stage of the EU ICT governance framework. The IDABC (Interoperable Delivery of Pan-European e-Government Services to Public Administrations, Businesses and Citizens) programme of the EU commission is one example of a

[22] In procurement, interoperability is sometimes considered in the context of technological neutrality and the functioning of markets. For a discussion, on these topics (*e.g.* in relation to consumer choice and risks of constraints on innovation), see Tsilas (2007).

measure intended to address such issues.[23] Another illustration is the e-GIF initiatives (e-Government Interoperability Framework) in the United Kingdom, which sets out the government's technical policies and standards for achieving interoperability and information system coherence across the public sector (www.govtalk.gov.uk). Other public-private initiatives focus on information exchange, such as the National High Performance Computing and Communications (HPCC) Software Exchange in the United States, which serves as a web-based resource for "promoting software sharing and reuse within and the HPCC community" including with respect to interoperability.[24] In this regard, promotion of interoperability for public use (*e.g.* in relation health or safety systems) or general welfare should be distinguished from policies that may distort the functioning of markets.

Achievement of interoperability

68. Software interoperability can be achieved in various complementary ways. The most common approaches being adopted by ICT companies include: *(i)* industry-community partnership and collaboration *(ii)* product design and testing *(iii)* sharing of technology and access to intellectual property (IP) and *(iv)* implementation of technology standards (BSA, 2007; Gasser and Palfrey, 2007).

- *Standards implementation* – Implementation of existing technology standards in products and services plays an important role in achieving interoperability by providing a stable technical solution to a common problem. There are various types of standards falling into two broad categories*: i)* There are "proprietary" standards developed and maintained by single vendor or a group of vendors, and *ii)* There are "open" standards which are developed through an open, voluntary, consensus-based process and publicly available to any interested party.

- *Industry-community partnership* – Industry and community partners – and sometimes competitors – collaborate at the domestic or international levels to share technical information with the purpose to define a common standard that may be used to develop interoperable software and services. They may sponsor standard working groups that consider customer needs and act to elaborate by building upon an existing standard.

- *Product design and testing* – In response to either the common standard established by industry-community or customer demand, developers sometimes move to elaborate interoperability through expanded product development activities throughout the design and testing phase of products.

- *Sharing common technology and access to intellectual property* – The adoption of shared technology or intellectual property may reduce the variability in the interface between different products and thus facilitate development of interoperable products in a quick and cost-effective manner.

The industry and government responses to the demand

69. Software interoperability has improved in the last decades despite the fact that complexity and variety of systems and new technologies have increased significantly. Major software industry firms are cooperating at an unprecedented level to align their technologies so that their products interoperate. In

23 IDBAC is engaged in a number of initiatives aimed to facilitate voluntary cooperation among EU member states to promote pan-European eGovernment interoperability. IDABC does not set the EC's official interoperability and standards policy. See: http://ec.europa.eu/idabc/ (last accessed 29.09.2008).

24 For more information on the HPCC software exchange, see: http://www.dlib.org/dlib/may98/browne/05browne.html (last accessed on 30.09.2008).

addition, there is increased engagement on the part of various stakeholders including ICT firms (*e.g.* network and systems management vendors), users and standards bodies. Through collaboration among interested parties, significant progress in software interoperability has been accomplished in recent years.

Collective efforts

70. Commercial software companies collaborate with other firms and also actively engage with broad-based standards bodies such as the World Wide Web Consortium (W3C), the Internet Engineering Task Force (IETF), the Web Services Interoperability Organization (WS-I), the European Computer Manufacturing Association (ECMA) and others (Box 1). They have been working together to address interoperability issues through standards development in technical committees of various organizations. One recent illustration of actions to promote interoperability is embodied in a group of standards based on the internet format XML (eXtensible Markup Language). XML is an open standard maintained by the W3C and many leading commercial software firms have contributed in the development of the basic XML architecture.

Box 1. Software related Bodies

The World Wide Web Consortium (W3C): W3C is an international consortium established in 1994. W3C's main mission is to develop Web standards and guidelines that ensure long-term growth for the Web. Since 1994, W3C has published more than 110 such standards, called W3C Recommendations. W3C has over 400 Member organizations, including vendors of technology products and services, and content providers, from more than 40 countries (www.w3.org).

ECMA International: ECMA is an industry association founded in 1961 and dedicated to the standardization of ICT and consumer electronics. The main activities of ECMA are to develop, in co-operation with the appropriate national, European and international organizations, Standards and Technical Reports in order to facilitate and standardize the use of ICT and consumer electronics. So far, ECMA has published more than 370 ECMA Standards and 90 Technical Reports, more than 2/3 of which have also been adopted as International Standards and/or Technical Reports (www.ecma-international.org).

European Information & Communications Technology Industry Association (EICTA): It was formed in 1999 and then in 2001, it merged activities with EACEM (the European Association of Consumer Electronics Manufacturers). EICTA set up an Interoperability Task Force in September 2003, and produced a "White Paper on Interoperability" in 2004 (EICTA White Paper 2004).

Organization for the Advancement of Structured Information Standards (OASIS) is a non-profit consortium, formed in 1993. OASIS acts as a driving unit to develop, and adopt standards of e-business and covers numerous areas including Web Services and e-Commerce (www.oasis-open.org).

Firm level efforts

71. Software developers are putting much emphasis on interoperability. For example, in early 2008, Microsoft announced four principles regarding interoperability: *ensuring open connection, enhancing support for industry standards, promoting data portability, and fostering more open engagement with customers and the industry, including the open source community* (www.microsoft.com/interop/principles). In building on these four principles, Microsoft will rely on inputs from the Interoperability Executive Customer Council, consisting mainly of Chief Information Officers or Chief Technology Officers of large enterprises and government agencies. Microsoft expects this initiative will make it easier and less costly for developers to create software that works smoothly with its current products. Also, Microsoft opened more than 30,000 pages of documentation about interoperability and APIs to the public.

72. IBM also announced recently that it is collaborating with nine business partners to help healthcare providers, clinics and hospitals improve productivity, increase quality and reduce costs through the use of Service Oriented Architecture (SOA). These partners are working to develop their latest

healthcare applications using the IBM SOA Foundation and supporting a set of open technology and industry standards.

Government-wide efforts

73. A European Parliament and Council Decision (2004/387/EC) promotes the Interoperable Delivery of Pan-European e-Government Services to Public Administrations, Businesses and Citizens (IDABC). The IDABC programme was launched on January 2005 for a period lasting until the end of 2009 in order to address problems that arise from ICT systems not being interoperable for both technical and organizational reasons. Interoperability is key to all of IDABC's horizontal eGovernment-related activities. It is expected that an integrated approach to interoperability will help to guide and support the development of electronic services. Activities to support interoperability have already been undertaken under IDABC's predecessor IDA II. These include the publication of the European Interoperability Framework (EIF 1.0), recommendations for promoting open document formats and an XML Clearinghouse feasibility study (www.ec.europa.eu/idabc). In addition to these efforts, the first *eGOV INTEROP Conference* was held in 2005 as a major means for confronting the progress of the various initiatives relating to eGovernment interoperability in Europe. The conference focused on all aspects of interoperability in eGovernment strategies, both from technical as well as from semantic, organizational and socio-economic perspectives, and discusses the progress of the various European initiatives in the field (www.egovinterop.net).

74. Japan announced new interoperability guidelines in 2007 (METI, 2007), which encourage Japanese government ministries and agencies to solicit bids from software vendors whose products support internationally recognized open standards. The Japanese government expects this new interoperability framework will propel healthy competition and open up more opportunities for small and medium-size companies in Japan

Issues surrounding interoperability

75. While many stakeholders – industry, government, and consumers – have agreed upon the need and benefits interoperability could bring, the scope, actual implementation or technical barriers that hinder it, remain somewhat controversial. One of the main controversies concerns the distinction between "open standards" and "open source". Open standards are vetted through an open process and can be seen as a set of rules and specifications that are based on the consolidated results of science, technology and experience, and aimed at the promotion of optimum community benefits (EICTA 2004). While, open source refers to software in which the source code is available to the public for use and modification from its original design.

76. Some public authorities have put considerable effort into the promotion of interoperability including initiatives in relation to open standards. In the European Union, for example, a program for implementation of interoperability known as IDABC (Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens) aims to facilitate the collaboration of public administrations in 27 different Member States and in EU institutions. Following the outlines of an eEurope Action Plan, an IDABC decision (Decision 2004/387/EC) was prepared "based on a framework of common principles and rules, as well as, on the agreement on open standards and interfaces for the implementation of interoperability between systems, applications, business processes and actors producing or using eGovernment services."[25]

[25] More on these points can be found on the following IDABC web pages at the following locations: http://ec.europa.eu/idabc/en/home and http://ec.europa.eu/idabc/en/chapter/5883.It is notable that one of the IDABC activities to achieve interoperability is "Support for Open Standards and Open Source

77. At the same time, some stakeholders stress that care should be taken to ensure technological neutrality and choice are maintained, in particular with respect to government actions in support of interoperability.[26] This can include an objective of seeking out and evaluating "best of breed" solutions based on the best value for the money, support, interoperability and compatibility (Lueders, 2005). A key concern is to avoid stifling potential innovative responses that may not yet be known to the public. Thus, there can be advantages in aiming for promotion of interoperability while maintaining an objective of technological neutrality in policy.

Summing Up

78. The scope of interoperability extends across the ICT sector and beyond. It is not just a single company issue, nor sector specific, nor country specific, but global. Although the benefit of interoperability is generally agreed and significant improvements have been achieved in recent years, there remain different views for its future evolution. Substantial progress towards interoperability is being achieved through market mechanisms. At the same time, further promotion of interoperability is being undertaken through various channels including private sector, government and multi-stakeholder initiatives. Success in this regard, may also have benefits for other types of software functionality such as mobility.

Accessibility

79. In view of the wide-ranging -- and expanding -- role of software in social, civic and economic life, software developers have been challenged to fight exclusion and ensure accessibility on a technical level for as much of society as possible.[27] The impetus for innovation in software accessibility is coming from a variety of angles including, among others, consumer demand, government mandates and procurement[28], international standards, and the corporate social responsibility of developers. Changes in demographics, including concerns related to population aging, as well as the opening of new technological possibilities (*e.g.* for expanded wireless communication) have raised awareness of the expanding needs and the options to address those needs (Reed, 2004). Significant progress has been made, but on-going technological change means that software accessibility issues will need on-going attention from developers and other stakeholders.

Software" http://ec.europa.eu/idabc/en/document/5313/5883. These Web pages were last accessed on 29.08.2008.

26 E.g., see Tsilas (2007) who argues that key elements in promoting interoperability should include: 1) protecting intellectual property; 2) avoiding technology mandates which stifle innovation and stunt economic growth and 3) promoting choice and technological neutrality in their procurement decisions and regulations. In another example, F. M. Buono and M. Sieverding argue for choice and technological neutrality in government in a web posting and newspaper article available here: http://www.metrocorpcounsel.com/current.php?artType=view&artMonth=February&artYear=2008&EntryNo=7853 (last accessed 29.09.2008).

27 The focus in this section is primarily on accessibility as a human-computer interaction issue that deals with the ability to use computer systems. Some observers interpret this concept more broadly as a social issue of access to ICT or a quest to close the so-called digital divide (*e.g.* see: http://www.internetworldstats.com/links10.htm, last accessed 08.08.2008).

28 For example, see TEITAC (2008).

Definitions and Benefits

80. Software accessibility can be seen as an issue centred on the possibilities for human-computer interaction. The International Organization for Standardization (ISO) defines accessibility as "*usability of a product, service, environment or facility by people with the widest range of capabilities."* The ISO definition can be compared to the definition from American National Standards Institute/Human Factors and Ergonomics Society (ANSI/HFES): "*The set of properties that allows a product, service or facility to be used by people with a wide range of capabilities, either directly or in conjunction with assistive technology. Although "accessibility" typically addresses users who have a disability, this concept is not limited to disability issue*s" (Gulliksen, 2004).

81. One of the primary objectives of software accessibility is to provide interfaces that may be used by a very broad range of people including those with disabilities[29] such as:

- *Visual*; low vision, lack of colour perception, blindness
- *Hearing*; hearing loss, deafness
- *Mobility*; restricted movement or control of arms, hands and figures
- *Learning or cognitive* impairment (*e.g.* dyslexia).

82. Enhanced software accessibility can benefit first and foremost those individuals dealing with such impairments. Yet, experience has shown that the potential benefits from improved accessibility can extend far beyond this population, contributing to general increases in productivity, reduced mental and physical stress, or economies in cost of training. For example, customizable fonts and color may enable users to pick settings that reduce eye strain and display more effectively in particular operating conditions. Keyboard navigation functions may enable users to move faster than using a mouse. Having alternative means of providing operator input may lower the risk of repetitive stress injury.

Innovative Technologies

83. The drive for accessibility plays an important role in promoting technological innovation and diffusion and exploitation of recent technologies and design strategies. The application of these enhanced approaches may open opportunities for organizations and businesses to reach new customers and markets or deliver more effective services.

84. In some cases, accessibility is facilitated through add-on assistive technology such as software or hardware that is used to increase, maintain, or assist the functional capabilities of individuals with disabilities. This technology offers alternative ways to access the contents on the monitor and to issue commands. It can be any technique that assists people in removing or reducing technical barriers and enhancing their activities. Examples of assistive technologies available:

- *Screen reader software* for those with visual disabilities, which operates by detecting and reading text displayed on the computer monitor using text-to-speech synthesis

[29] More information on the relationship of impairments, technology and software accessibility can be found on the website of the Royal National Institute of Blind People here: http://www.rnib.org.uk/xpedio/groups/public/documents/PublicWebsite/public_sactypes.hcsp.

- *Voice recognition software* that allows a person to simulate typing on a keyboard or selecting with a mouse by speaking into the computer.

- *Screen magnification software* that allows a low-vision user to read more easily information displayed on a monitor.

- *Comprehension software* that allows a dyslexic or learning disabled person to see and hear text as it is manipulated on the computer screen.

- *Keyboard enhancements and accelerators* such as StickyKeys, RepeatKeys, BounceKeys.

- *Software mouse simulators* that allow users to move the mouse pointer by pressing keys on the numerical pad.

Government polices

85. In recent decades, governments in many OECD countries have increased the priority they place on this issue. Many have moved to encourage enhanced software accessibility via voluntary guidelines, awareness raising campaigns, procurement practices, legislation or standards. For example, in the late 1990s, the United States government amended Section 508 of the Rehabilitation Act requiring Federal agencies to purchase electronic and information technology that is accessible to people with disabilities.[30] The law applies to all Federal agencies when they develop, procure, maintain, or use electronic and information technology. These agencies must ensure that this technology is accessible to employees and members of the public with disabilities to the extent it does not impose a burden. Section 508 speaks to various means for disseminating information including computers, software, and electronic office equipment. It applies to – but is not solely focused on – Federal government pages on the Internet or the World Wide Web.

86. The European Union provides another example. In 1999, the European Commission proposed an initiative called eEurope. The main purpose of this initiative was to bring the benefits of the information society to all Europeans. The Commission followed up this initiative in a communication of 2001 stating that one of the specific objectives of the eEurope Action Plan is to improve access to the Web for the 37 million people with disabilities in Europe as well as for the growing number of older persons. In 2005, the Commission adopted a further Communication on eAccessibility aiming to ensure that as many people as possible can fully participate in the Information Society. In particular, the initiatives encouraged the ICT sector and public bodies to better take into account the needs of the elderly and people with disabilities. Moreover, the Commission is promoting adoption of the World Wide Web Consortium's (W3C) *Web Content Accessibility Guidelines* for all public Web sites in European institutions and member states.

87. Japan, as well, has been actively involved in accessible ICT initiatives, contributing a number of proposals to the ISO and cooperating with other Asian countries. The Japan Electronics and Information Technology Association is responsible for developing industry standards that will foster a digital network society aimed at improving quality of life via ICT advancement. In 2003, the Japan Industrial Standard Committee (JISC) issued the standardization policy addressing needs of the elderly and persons with disabilities, which include establishment of a systematic standardization scheme, prioritizing of standardization fields including via annual updates, and encouraging more international cooperation (Yoshida, 2006).

30 Further information on this legislation can be found here: http://www.section508.gov/.

Guidelines and standards

88. There is a wide range of resources available to software developers seeking guidance in designing accessible software-user interfaces. A number of standards or guidelines have been published by international organizations, governments, non-governmental organisation and industry associations. Various supplementary resources are available from bodies such as ISO and W3C, private entities like Apple, IBM, Microsoft, the Mozilla Foundation and Sun, and academic institutions such as the Trace Centre at University of Wisconsin-Madison.[31] These include, for example, academic studies, design guidelines, training materials and tools. Cooperation among different stakeholders has also been active. The proposed US Software Accessibility standard was contributed to ISO in 2000, and served as a basis for ISO Technical Report 16071 – Guidance for Software Accessibility (Reed 2004). In Asia, the Japan-Korea-China Accessibility Design Committee was established in 2003 to promote standards cooperation among standards bodies.

89. Although there are many different guidelines that reference software accessibility, they tend to share some common features. For example, they generally address one or more categories of software accessibility issues (Kavcic, 2005):

- System and content, including object information and timing,
- Keyboard access and pointing devices
- Sound and multimedia,
- Display
- Verification of accessibility and documentation

90. Guidelines for Web accessibility are often also directly relevant to software accessibility, albeit with more specific treatment of the Internet dimension (Brewer, 2004). For example, the W3C Web Accessibility Initiative has delivered a series of guidelines (Box 2).

Box 2. W3C Web Accessibility Guidelines

Web Content Accessibility Guidelines (WCAG): WCAG 1.0 became a W3C Recommendation in 1999. Since then, many organizations and private sector firms have adopted the Guidelines. WCAG is composed of general guidelines and specific checkpoints against which conformity can be evaluated. The checkpoints are divided into three priority levels, addressing: i) absolute barriers, ii) significant barriers, and iii) additional accessibility support for disabled people. The development of WCAG 2.0 is now underway, taking into account extensive feedback on WCAG 1.0 and advancements in Web technology.

Authority Tool Accessibility Guideline (ATAG): This guideline became a W3C Recommendation in 2000. It incorporates guidelines on how to make software applications that are used to create Web sites accessible to disabled people and ways to support the production of accessible Web sites. It includes a checklist to ask software vendors when evaluating accessibility support in their Web authoring tools.

User Agent Accessibility Guidelines (UAAG) : UAAG became a W3C Recommendation is 2002. It can be considered as complimentary to WCAG, since software used for accessing the Web also needs to be accessible. UAAG explains how to make browsers and media players accessible to disabled and how to make them interoperate smoothly with assistive technologies.

For more information see the W3C web site: http://www.w3.org/WAI/ .

31 For example, see the Trace Center web site: http://trace.wisc.edu/ .

Company efforts

91. Many software firms are committed to making their products broadly accessible, including users with disabilities or impairments that occur with aging. Most of them look to accessibility best practices and standards defined by guidelines such as Section 508 of the U.S. Rehabilitation Act and the W3C WAI. Several illustrative cases are presented below:

- Microsoft, for example, has publicly declared its commitment to accessibility with the adoption of its accessibility policy and increased its accessibility efforts in 1995. Following the announcement, accessibility features were built into the Windows 95 operating system. Since then, a continuous effort has been made to improve accessibility in subsequent versions of the operating systems, incorporating new features such as magnifiers, a text-to-speech utility, adjustable font sizes and colours and an on-screen keyboard.[32] Besides developing assistive technologies, the firm has also made efforts to raise awareness, provide design guidance, and run accessibility resource centres to demonstrate available accessible technology solutions.

- IBM formed a worldwide Accessibility Center with locations in the U.S, Europe, Japan and Australia in 2000.[33] The Center aims to foster product accessibility by working toward the harmonization of worldwide standards and applying research technologies to solve problems experienced by people with disabilities; it creates industry-focused solutions and generates accessibility awareness. As part of the IBM research organization, the Accessibility Center has a direct line to the scientists developing new technologies.

- In 2001, SAP has established an Accessibility Competence Center (ACC). The ACC is the focal point for the SAP Accessibility Program which aims to ensure the firm meets or exceeds compliance with accessibility requirements outlined in legislative and industry standards.[34]

Summing Up

92. There has been considerable progress in addressing software accessibility issues as highlighted in various examples above. However, technological change and progress mean that the work to enhance accessibility remains on-going and that new opportunities may arise to better response to accessibility needs. In recent decades, various institutions – public and private – have developed to ensure a continued focus on this work and to assist in the diffusion of best practices.

Reliability

93. With the world becoming more and more dependent on software, software failures can cause more than mere inconvenience. Today, software errors or failures can expose society to the risks of critical infrastructure failures, severe economic loss or even human fatalities. Where they occur, reliability problems can originate from a variety of sources ranging from poorly designed user interfaces to direct programming errors or improper implementation. In view of the increasing complexity of software, efforts

---

32 More information is available on the Microsoft web site, here: http://www.microsoft.com/enable/microsoft/.

33 More information is available on the IBM web site, here: http://www-03.ibm.com/able/accessibility_services/index.html.

34 For further information, please see the SAP web site here: http://www.sap.com/platform/netweaver/standardssupport/accessibility.epx .

to address reliability needs now engage the full range of stakeholders including developers, vendors, users and others.

94. Reliability is part of a cluster of inter-related concepts linked to the ability of users to have confidence that the software products they employ will work consistently and not in an unpredictable manner. Software experts apply a variety of terms and definitions in describing various perspectives on this cluster of issues, such as reliability, quality assurance and dependability. These terms can be defined as follows:

- *Software reliability* -- "The probability of failure-free software operation for a specified period of time in a specified environment." (ANSI, 1991 and Musa *et. al.*, 1982, as cited in Lyu, 2007)

- *Assured software* -- "Software that has been designed, developed, analyzed and tested using processes, tools, and techniques that establish a level of confidence in its trustworthiness appropriate for its intended use." (CNSS, 2006)[35]

- *Dependability* -- "The trustworthiness of a computing system which allows reliance to be justifiably placed on the service it delivers", which includes as special cases such attributes as reliability, availability, safety, security. (IFIP, 1988)

95. As Michael Cusumano – a leading software sector expert based at MIT – points out, successful software development is a creative process requiring a balance between establishment of enough structure to keep projects under control, but not so much as to stifle creativity and flexibility (Cusumano, 2004). Despite significant progress in software development approaches in recent decades, he notes that problems have persisted. For example, he points to observations that as many as 75 to 80% of software projects are typically late and over budget.

96. Concerns about reliability are heightened by user perceptions. A survey of business enterprises in Japan (Nikkei, 2003) found that 73% of system development projects fail to deliver as expected in one or more key dimensions specified as "quality", "cost" or "delivery". The respondents pointed to the increased costs or delays associated with these shortfalls.

97. To some extent, the problems of achieving reliability in software systems reflect the difficulties of programming. As Cusumano (2004, p. 132) notes, "Writing program algorithms is usually not a routine activity. It generally involves creativity and invention on some level, as well as problem solving and trial and error." Moreover, "In custom software projects, users often do not know what they want until they see part of the system in front of them." The complexity of the programs and the environment ensures that achievement of reliability is a clear challenge.

98. Attempts to improve software reliability can be applied in different parts of the development process, which might be frames broadly as including requirement specification, design, programming, testing, and run time evaluation. Donzelli *et al.* (2006) note that dependability requirements are particularly difficult to deal with because they cover many different aspects of a system at the same time. Moreover, such requirements are deeply rooted in the specific context. Different stakeholders will focus on different attributes such as availability, catastrophic failure avoidance, and deliberate-intrusion prevention, and also differ in the definitions of these attributed. In some areas such as defence or nuclear energy or automotive

[35] A related issue concerns information assurance, a concept generally linked to security and integrity of data. For a discussion of the relationship to quality, see: Voas, J., and Wilbanks, L. (2008), *Information and Quality Assurance An Unsolved, Perpetual Problem for Past and Future Generations*, IT Pro, May/June.

safety, zero failures may be tolerated. In others, the objective may be 'reasonably low failure rates". Given these considerations, it is not surprising that Donzelli *et al* note that erroneous or omitted requirements are often indicated as the main reasons for project failures.[36] A similar view was echoed by a survey in Japan of firms purchasing software systems; among firms that experienced failures of projects to deliver expected results, the leading cause – cited by 36% of respondents – concerned requirement definitions (Nikkei, 2003).[37]

99. While a number of approaches and tools have been developed to reduce the likelihood – in advance – of reliability issues arising.[38] Developers employ design and code reviews to maintain quality controls as projects advance. Software reliability can be enhanced during the process through techniques such as fault prevention, fault tolerance, fault removal and fault forecasting[39] (Avizienis, 2001). However, reliability issues invariably arise and developers rely on applied testing of software as a primary technique for detecting them so they can be tackled. Developers employ such testing in an on-going manner during the development process with respect to such dimensions as the functioning and the stability of software. This applies to both the individual modules being developed and the interaction between modules. Issues are then addressed as they come to light through the testing procedures.

100. There can be numerous ways to conduct tests of the reliability of software using both human testers and automated tests. Two traditional approaches are black box testing and white box testing. Black box testing treats the software as a black-box and is implemented without requiring knowledge or understanding of internal structure of the software. White box testing is used when the tester has access to the internal data structures, code and algorithms and takes these factors into account. Under one common practice in software testing, an independent group of testers may review the product after it is developed and before it is delivered to the customer. Alpha testing is performed by potential external users or by an independent test team at the developers' site. Then follows Beta testing during which the so-called "Beta Version" is released to an independent group of users. Sometimes, the Beta versions are made available to the public in order to increase the volume of feedback.

101. The challenges in software reliability stem in part from the complexity and difficulty of analysing software applications in various domains. Thus, it is not surprising that the testing procedures can be extremely labour-intensive. In view of the complexity and difficulty in anticipating all the potential bugs, in some cases developers may be reluctant to spend too much effort in this area; the cost-effectiveness of additional effort may not be clear (Lyu, 2007). That is, the cost of reliability can be a direct function of the cost of testing and it can be very expensive. When the product delivery schedule is tight, reliability may receive lower priority. In addition, although many companies are collecting failure data, it is often the case that they will not share the data or experiences, which makes comparing or benchmarking reliability results difficult.

102. On the other hand, as Cusumano (2004, p. 177) points out, failure to address reliability issues (and quality issues more broadly) at the testing stage can entail massive costs due to product recalls and service provision. In the case of critical systems (such as safety-related systems), the costs of failure may

---

36 One of the approaches that address software dependability is the Unified Model of Dependability, which supports stakeholders in formulating their requirements and mapping them effectively (Donzelli *et al.*, 2006).

37 This is based on the sample of 498 companies that provided valid responses.

38 For example, according to IEEE (2004), IEEE Standard 982.1-1988 -- *IEEE Standard Dictionary of Measures to Produce Reliable Software* -- provides a set of measures for evaluating the reliability of a software product and for obtaining early forecasts of the reliability of a product under development.

39 However, it remains difficult to measure and model software reliability (Pan, 1999).

be even larger. Hence, many developers invest enormous amounts in efforts to ensure an appropriate degree of reliability. While the relationship between reliability enhancement and this investment is not necessarily one-to-one (i.e., because problems vary in the degree of difficulty they pose), it can help. As Cusumano advises software entrepreneurs, "You can never do too much testing."

103. Many developers have mechanisms to solicit feedback from users even after release of the software, which provide input for revisions aimed at improving the reliability of a software product in future updates and releases. In some cases, failure reports are generated automatically via built-in routines while in others there are easy access links to facilitate reporting of bugs.

Summing up

104. Software reliability can be viewed from various perspectives and must be considered in context. Software developers have learned a lot over the last few decades about approaches to enhance reliability of software in line with expectations and market demand, and here as well there have been innovations (e.g., in methods). Nevertheless, reliability and related software qualities can be expensive and there can be tradeoffs between enhanced reliability and the cost of software.

Box 3. User participation in the development of software functionality

In many sectors, the engagement of users in product development processes is a practice that is still only in its infancy. In the software sector, however, user-driven innovation is already widespread and continues to evolve. Indeed, the software industry can be considered as being among those on the leading edge of this type of development.

The active participation of users in the development of software is facilitated by the very nature of software as a digital, intangible product. For example, these characteristics enable real-time, global transmittal of content, insertion of incremental changes in existing products and automation of certain user input provision (*e.g.* software failure reports). Users are contributing concepts and even software code that goes beyond what otherwise would have been available. As a consequence, many software developers are able to produce products with a competitive edge. The types of advantage may vary by product, but can potentially include better time-to-market, quality, cost or functionality, among other characteristics, in comparison with products developed using more traditional approaches.

The manner of engaging user input may vary widely, from passive monitoring of usage and behaviour to more active engagement at the initiative of either the users or software developers. The resulting input may be handled in a closed proprietary manner between the parties or shared broadly via more open approaches. One common approach to encouragement of this type of user engagement is for developers to circulate alpha or beta versions of software, in some cases via the internet; leading users or early-adopters of the provisional technology can then make a contribution via feedback or other input. Another approach is "crowdsourcing", whereby problems are posed via closed networks or the Internet to various communities in an open manner in order to draw on a wide range of expertise.

OECD (2007) presents the concept of the "participative web", which is based on "an Internet increasingly influenced by intelligent web services that empower users to contribute to developing, rating, collaborating and distributing content via the Internet. This concept is quite relevant in the case of software as users increasingly engage via the Internet in the process of software development in a variety of ways such as articulating their need and suggestions for enhanced functionality, creating their own applications via web-based platforms (discussed in Chapter 1), and contributing specific elements as input to external software developers (*e.g.* building on existing applications).

New web software tools enable users, even individual non-technical users, to contribute actively. Often, the user-driven input is facilitated by creative approaches to communication among the various stakeholders (*e.g.* using web-based platforms) and among various software applications via open web standards and interfaces. Thus, the so-called "collective intelligence" of the Internet users can be a creative force in software development including such well-known products as the Linux operating system, Firefox browser or Apache server software.

Incentives for users to engage in innovation processes operate with respect to businesses and individuals. For businesses, the incentives may be direct remuneration (e.g., as when they sell applications via platforms such as software.com) or indirect benefits through improved software functionality that enhances their own products or processes (i.e. leading to increased revenues or decreased costs). For individuals, the incentives may be monetary (e.g. payments, licensing fees or voluntary contributions) or quasi-non-monetary (e.g. recognition, enhancement of reputation or experience gained (In the literature, this latter type of incentive has sometimes been cited as "intrinsic motivation" Bitzer *et al.*, 2007).

ANNEX
BEYOND THE TRADITIONAL SOFTWARE SECTOR: THE ROLE OF BUSINESSES AS SOFTWARE DEVELOPERS AND USERS[40]

While much of the buzz about software in the popular press is focused on consumers, businesses beyond the traditional software sector are playing a very significant role as both users and developers of software. Firms across the economy are deploying large amounts of software to power their operations and products. As discussed in Chapters 1 and 2, the amount of software in many types of products is increasing rapidly. For example, the number of lines of source code in a mobile phone is expected to increase from 2 million today to 20 million by 2010; by that time, a car may contain 100 million lines of code (Charette, 2005; also Ito, 2007). The growth of software involves far more than a technical evolution. It will have consequences for companies, industries and national economies.

One indication of the extent of the interrelationship between the software sector and other industries can be found in the composition of employment by sector. Table A.1 presents data for the European Union on the share of computer specialists in total employment of various sectors, as calculated drawing on European labour force survey data. As might be expected, the top-ranked industries are "computer and related activities" and "manufacture of office equipment and computers". Perhaps more notable is the engagement of significant numbers of computer specialists (which includes software professionals) across a wide range of sectors, from collection, purification and distribution of water to post and telecommunications. Among the NACE 2-digit sectors shown in the table, all have 1.4% or more of their employment in the occupational classification of computer specialists. A similar measure is presented for the United States in Table A.2, though at a higher level of sectoral aggregation. Here again, the presence of computer professionals, including software professionals, is notable across the economy. Only 4 of the 13 sectors presented have less than 1% of total employment in the computer professional category. Business and professional services account for the largest share of both computer-related and software-related employment, followed by the information sector, financial activities and the public sector.

[40] This section draws on an issues paper prepared for the OECD by Prof.dr.ir. Michiel van Genuchten (2008), as well as other sources.

Table A.1. Top 25 ranking of industries according to their share of computer-related employment in total employment, EU15, 2006

EU15 - NACE Sector	%
72 Computer and related activities	53.5
30 Manufacture of office machinery and computers	23.4
32 Manufacture of radio, television and communication equipment and apparatus	6.8
73 Research and development	5.2
66 Insurance and pension funding, except compulsory social security	5.1
64 Post and telecommunications	4.7
65 Financial intermediation, except insurance and pension funding	4.5
33 Manufacture of medical, precision and optical instruments, watches and clocks	3.7
99 Extra-territorial organizations and bodies	3.1
40 Electricity, gas, steam and hot water supply	3.1
67 Activities auxiliary to financial intermediation	3.0
11 Extraction of crude petroleum and natural gas; service activities incidental to oil and gas extraction excluding surveying	2.7
35 Manufacture of other transport equipment	2.3
62 Air transport	2.1
31 Manufacture of electrical machinery and apparatus n.e.c.	2.0
24 Manufacture of chemicals and chemical products	1.8
22 Publishing, printing and reproduction of recorded media	1.8
74 Other business activities	1.7
34 Manufacture of motor vehicles, trailers and semi-trailers	1.6
41 Collection, purification and distribution of water	1.6
21 Manufacture of pulp, paper and paper products	1.6
71 Renting of machinery and equipment without operator and of personal and household goods	1.5
75 Public administration and defence; compulsory social security	1.5
29 Manufacture of machinery and equipment n.e.c.	1.4
23 Manufacture of coke, refined petroleum products and nuclear fuel	1.4

Note: The ISCO categories "213 computing professionals" and "312 computer associate professionals" were used as a proxy.

Source: Authors' calculations based on the European Labour Force Survey.

Table A.2. The share of computer-related employment and software and programming specific employment in total employment, USA, 2004

Industry	Computer related (%)	Software + programming (%)
1 Agriculture, forestry, fishing, and hunting	0.2	0.2
2 Mining	1.9	1.2
3 Construction	0.1	0.0
4 Manufacturing	2.6	1.2
5 Wholesale and retail trade	1.3	0.5
6 Transportation and utilities	1.4	0.5
7 Information	8.8	3.4
8 Financial activities	3.8	1.6
9 Professional and business services	9.3	4.0
10 Educational and health services	1.1	0.2
11 Leisure and hospitality	0.4	0.1
12 Other services	0.6	0.2
13 Public administration	3.6	1.2
Total	2.4	0.9

Note: The CPS categories "110 computer and information systems managers", "1000 computer scientists and systems analysts", "1010 computer programmers", "1020 computer software engineers", "1040 computer support specialists", "1060 database administrators", "1100 network and computer systems administrators", and "1110 network systems and data communications analysts" are used. The ISCO categories "213 computing professionals" and "312 computer associate professionals" are used as a proxy.

Source: Authors' calculations based on US Current Population Survey (CPS).

Implications at company level

The growth in the volume of software needed to deliver the functionality demanded in products across the economy poses challenges to producers beyond the traditional software sector. To respond using in-house development approaches, typically entails significant growth of the software engineering and support resources. Companies must assess whether to attempt such development keeping in mind the cost of building such capacity and the challenge of finding the necessary qualified staff (*e.g.* software engineers).[41]

If the software is to be developed in-house for an embedded application, then the projects are generally financed from existing operations. Some companies have margins or market shares that allow them to bear the increasing software costs without excessive problems. Examples can be found in specific high-volume consumer electronics products (where economies of scale can be reaped) or some high margin medical devices. For such firms, there may be strategic advantages or relatively lower costs in taking the in-house development approach. However, in other cases, the software development burden would be excessive for a single producer. The computer industry faced this over recent decades. It may be on the horizon for individual car manufacturers, who may lack in-house capacity for development of the many large or specialized applications that are foreseen (*e.g.* guidance or safety systems). Thus, the choice of how to proceed with software development is challenging an increasing number of manufacturers and service-sector firms on an on-going basis.

This choice is not always an easy one, particularly because management perceptions of investment in software development may vary. In cases where software is not generating a separate revenue stream, it may be viewed by management as an operating cost – one that may be growing, and something to be minimized. On the other hand, the entry of new suppliers of such software on the market for inputs or the entry of competitors to the manufacturer on the consumer market for the product may signal that a given sector is becoming more dynamic. Management may then perceive a strategic or tactical advantage in engaging more deeply in the software development process including via increased openness and collaboration with partners beyond the walls of the firm or even launching a new software venture.

[41] There is strong demand across the economy for qualified software professionals, and some manufacturers – where software may be viewed as a sideline – may be at a competitive disadvantage (*e.g.* in terms of image as perceived by software professionals) vis-à-vis leading software or computer service firms in terms of recruitment.

Box . Embedded Software: Illustration of the Successful Launching of An Affiliated Producer

Starting an affiliated software business will change the economics of a parent firm that formerly produced embedded software in-house, potentially in a beneficial way.

In shifting the embedded software activity to a new affiliate, additional costs will be incurred establish the marketing and sales capacity for the software formerly sold with the hardware, but now brought independently to the market. On top of that, the legal infrastructure has to be built to protect the software and the associated intellectual property.

How might the economics work in a successful case? Let us assume that half of the costs in building a software business are related to developing and maintaining the software and the other half are related to the business aspects of software, such as marketing, sales and legal expenses. For the parent firm, bringing the software to the market via a separate entity means doubling the costs in this case. On the other hand, if the software affiliate is then successful in selling the software to its mother company as well as to, say, two other companies then it can begin to reap some offsetting economies of scale. Consider the comparison in the Box Table.

Box Table. Economics of software production in-house versus an affiliate

In-House Production		Shift to Sourcing via an Affiliated Software Vendor	
Sales (marketed as part of the hardware products)	0	Sales: To parent company To two other companies (for same price)	 75 150
Cost: software development	100	Costs: Development: Software marketing and sales:	 100 100
Result	-100		25

In this example of a successful case, the parent company obtains the software it needs for 25% less than if it had developed the software internally and its affiliate earned a profit of 25. By increasing the volume of production, the unit costs were reduced and the parent firm benefitted directly and via the affiliate. In this way, software started to create value for the company involved.

Implications for industries

The growth of software in terms of size, complexity and range of application poses challenges to existing firms. In cases where firms can consolidate fragmented demand across markets and supply more standardized software solutions, there is the potential them to introduce disruptive and beneficial change. The computer industry was completely changed in the 1980s and 1990s as a result of opening of systems to permit independent software supply. As an article in *The Wall Street Journal* pointed out (WSJ, 2005),"Until the 1980s, a handful of giant manufacturers controlled the design, construction and sale of their machines, along with most of the software that ran on them. But then came along the personal computer, which relied on standardized chips and software." There are indications that similar revolutions may be set to happen in other industries.

Volume is a key element in meeting the growing software demands. The digital, intangible nature of software implies low marginal costs. This can enable software suppliers to have better returns overall and per unit in cases when volume is large. Moreover, the larger software companies have accumulated

significant software R&D capabilities that can be put to use in other industries that are struggling to meet the need for software. In view of such factors, this section considers three illustrative industry case studies where software may prove transformative in coming years: automobiles, mobile phones and healthcare.

Software growth in the automobile industry

Software is playing an increasingly important role in delivery of the functionalities sought by automotive customers, with the result that the size of programs is growing enormously. If projections are accurate, then the software bundle for a fully-equipped car may grow from about 1 million lines of code in 2000 to some 100 million lines of code (about twice the size of Microsoft's Vista operating system) by 2010. At the same time, individual car producers control just a fraction of the market and will have trouble to mobilise the sales volume needed to justify the investment in development of such massive amounts of code in house.[42]

As a result, it is not surprising that auto manufacturers are already reaching out to buy-in the necessary functionality. In some cases, this is happening indirectly through external purchases of hardware units with embedded software such as multimedia devices and navigation systems. These are, however, rather isolated pieces of hardware. One response can be seen in the Japan Automotive Software Platform Initiative (Jaspar), which is aiming to pool resources from across the Japanese auto industry to address automotive software challenges in a manner that yields economies of scale.[43] From outside of the sector, major software suppliers have targeted the automotive sector as a market for development. For example, Microsoft's *Windows Automotive* is an open platform aiming to facilitate third party developer participation in creation of automotive software solutions.[44]

Liability issues remain to be resolved and may impede opening and development of some aspects of automotive software. Given that much of the software in a car relates to safety and critical operational features, it is more closely integrated than, for example, a multimedia system. This may impede the opening of some embedded systems to independent outside software suppliers.

Software growth in the mobile phone industry

The number of mobile phones sold in 2008 in the world is estimated to be about 1.3 billion. The size of the software bundle in these products is growing rapidly and is slated to increase from an estimated 2 million lines in 2006 to 20 million in 2010. The volume leaders in the mobile phone industry are the large technology firms such as Nokia, Samsung and Motorola.[45] Nokia alone produces over 500M mobile phones a year.

---

42 Toyota and General Motors each sold approximately 9.37 million cars in 2007 (BBC news, 2008), in an overall market of approximately 60 million cars.

43 As stated on the Jaspar web site, "JasPar will strive to reduce technology development costs and promote technology development by encouraging Japanese companies to collaboratively develop pre-competitive technologies such as automotive LAN enabling technology, middleware and software platform." Available here: https://www.jaspar.jp/english/guide/purpose.php, (last accessed on 07.08.2008).

44 A factsheet on Window's Automotive is available here: http://download.microsoft.com/download/f/b/5/fb5efead-ef87-4ddc-a05b-2d75154e0edc/WA_Datasheet.pdf (last accessed 07.08.2008).

45 In the second quarter of 2008, the respective market shares as posted on Imran's Everything Cellular web site are: Nokia (41.1%), Samsung (15.4%) and Motorola (9.5%), available at: http://www.mobileisgood.com/statistics.php#current (last accessed: 07.08.2008).

The amount of software in high-end phones or smartphones is increasing especially rapidly. (A smartphone is typically defined as a mobile phone with an open operating system.) Moving to smartphones can be seen as a way in which the mobile phone industry tries to handle the increasing amounts of software required to deliver the desired functionality in their products. By providing a more open operating system, it allows multiple software companies to supply the ever-increasing amounts of software required providing a competitive phone. Dominant operating systems in mobile phones today are Symbian and Windows Mobile. It brings the mobile phone industry from the embedded stage into the open system stage.

Two recent events indicate that a portion of the market for some types of software in the mobile phone industry may shifting towards open source approaches. First, in October 2007, Google announced its platform Android, which it claims will be the first complete, open and free mobile software platform (OHA, 2008). Then, in June 2008, Nokia announced its purchase of the Symbian operating system with the intention to bring the system into the open source domain. Many of the key players in the mobile phone industry have engaged in the Android or Symbian initiatives.

One accelerator for the changes in the mobile phone market may be the fact that only five manufacturers supply the vast majority of mobile phones. They have clear volume leadership and may be in a position to agree upon standards that facilitate software development. Moreover, the expanding technological capacity of the phones is making it possible to add new functionalities; nowadays, a vast amount of software required to produce a competitive mobile phone.

Software growth in the healthcare industry

The aging population around the world is providing impetus for improved software functionality. As with cars and mobile phones, the amount of software in healthcare electronics products is growing rapidly especially for medical diagnostics equipment and information management systems (storage and exchange) in medical institutions. ITEA[46] (2005) estimates that R&D expenditure for software in the medical equipment industry will grow from 7 billion Euros in 2002 to 28 billion Euros in 2015. Software R&D is slated to increase its overall share of R&D in the sector from 25% to 33%. Moreover, ITEA estimates that software staff may account for up to 60 percent of the R&D staff for major providers.

In relation to information management, one of the drivers for software demand in the sector is the expansion of work on human DNA. New insights concerning DNA are enabling more personalised healthcare (*e.g.* different therapies for different people with the same disease), but also requiring availability of detailed information about a patient's DNA and healthcare history.

In the past, many medical systems employed closed embedded systems, but the growth in demand for software is leading equipment manufacturers to consider ways to collaborate and leverage their efforts. One factor that may impede this development is, again, concern about liability issues. Similarly to the automobile industry, some medical equipment suppliers may opt to keep control of critical aspects of the software in order to contain the risk. The role of regulatory bodies may also impede the growth of small software companies in this business, in view of sometimes lengthy and expensive approval procedures.

Summing Up

The evolution of software is impacting industries across the economy and appears set to transform sectors such as the car, mobile phone and healthcare industries. Software is becoming increasingly

[46] ITEA stands for Information Technology for European Advancement is "a strategic pan-European programme for advanced pre-competitive R&D in software for Software-intensive Systems and Services". For more information see: http://www.itea-office.org/ (last accessed: 07.08.2008).

mainstreamed in the products and services of modern life. In many instances, the markets for such software remain fragmented with many firms still developing software internally or opening their systems only partially to outside collaboration or suppliers. Liability concerns may play a role in this regard. On the other hand, there are tremendous possibilities for economies of scale in software and firms are striving to find ways to consolidate the growing demands for software and thereby reap the benefits.

REFERENCES

Anderson, R., Böhme, R., Clayton, R., and Moore, T. (2008), *Security Economics and the Internal Market*, ENISA, Heraklion.

Anderson, R., and Moore, T., (2006), "The Economics of Information Security", *Science* 314 (5799), pp.610–613.

ANSI (1991), "Standard Glossary of Software Engineering Terminology", STD-729-1991, ANSI/IEEE, 1991, as cited in Pan, Jiantao (1999).

AusCERT, Australian High Tech Crime Centre, Australian Government Attorney General's Department (2007), « Australian Computer Crime and Security Survey », AusCERT, Brisbane, www.auscert.org.au/render.html?it=2001.

Avizienis A., Laprie JC, Randell B. (2001), "Fundamental Concepts of Dependability", Technical report, LAAS-Newcastle University-UCLA.

Ballou, M. (2008), Facing the Software Quality Challenge, presentation based on ICD (2008), Debugging and Business Value Survey, April.

BBCNews (2008), GM and Toyota Level on 2007 Sales, http://news.bbc.co.uk/2/hi/business/7205073.stm, (last accessed: 07.08.2008).

Bennett, C., Raab, C. (2006), *The Governance of Privacy: Policy Instruments in Global Perspective*, MIT Press, London.

BERR (Department for Business, Enterprise and Regulatory Reform) (2007), « Information Security Breaches Survey », BERR, London, www.berr.gov.uk/sectors/infosec/infosecdownloads/page9935.html.

BIAC/ICC (2003), *Information Security Assurance for Executives*, BIAC/ICC, Paris.

BIAC/ICC (2004), *Securing your Business: Information Security Issues and Resources for Small and Entrepreneurial Companies*, BIAC/ICC, Paris.

Bitzer J., Schrettl, W., and Schroeder, P. J. H. (2007), "Intrinsic Motivation in Open Source Software Development", *Journal of Comparative Economics*.

Braunschweig B. (2005), "Software Interoperability for Petroleum Applications", Oil & Gas Science and Technology – Rev. IFP, Vol 60 (2005), No. 4.

Brewer, J. (2004), "Web Accessibility Highlights and Trends", Proceedings of the 2004 international cross-disciplinary workshop on Web accessibility (W4A), referenced here: http://portal.acm.org/citation.cfm?id=990657&dl=GUIDE&coll=GUIDE .

BSA (2007), BSA Statement on Interoperability: Innovation, Choice, and the Role of Governments, Business Software Alliance, Brussels, accessed on 30 April 2008 at: www.intgovforum.org/Rio_Meeting/interventions/.

Charrette, R. N. (2005), "Why Software Fails," *IEEE Spectrum*, September.

CNSS (2006), National Information Assurance Glossary, CNSS Instruction No. 4009, Committee on National Security Systems, US National Security Agency, Ft Meade MD 20755-6716, available on-line at: http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf (last accessed on 9 July 2008), June.

CSI (Computer Security Institute) (2007), « CSI Computer Crime and Security Survey », CSI, www.gocsi.com.

Cusumano, M. (2004), *The Business of Software*, Free Press, New York.

Department of Justice, Victoria, Australia, (2002), "Web seals of approval", www.consumer.vic.gov.au/CA256902000FE154/Lookup/CAV_Publications_Reports_and_Guidelines/$file/webseals_of_approval.pdf.

Dewar, R., Schonberg, E. (2008), « Computer Science Education: Where Are the Software Engineers of Tomorrow? », Crosstalk, www.stsc.hill.af.mil/CrossTalk/2008/01/0801DewarSchonberg.html.

Donzelli P.,Shull F., Asgari S. and Basili V.(2006) "Evolving a Dependability Requirements Elicitation and Modeling Framework Based on Use", Technical Report CS-TR-4851, University of Maryland, available on line at: https://drum.umd.edu/dspace/bitstream/1903/4026/1/UMD+class+experiment+-+UMD+tech+report+Final+Copy.pdf (last accessed on 9 July 2008), November.

Dynes, S., Goetz, E., Freeman, M. (2008), « Cyber Security : are incentives adequate? », in IFIP International Federation for Information Processing, Volume 253, Critical Information Infrastructure Protection, Springer, Boston, pp. 15-27. http://mba.tuck.dartmouth.edu/digital/Research/AcademicPublications/CriticalInfrastructure.pdf.

EICTA (2004), *Interoperability*, White Paper, Belgium.

ENISA, « Information Security Awareness Programmes in the EU », ENISA, Heraklion, www.enisa.europa.eu/doc/pdf/deliverables/enisa_is_aw_programmes_eu.zip

European Committee for Standardisation (CEN) (2005), "Analysis of Privacy Protection Technologies, Privacy- Enhancing Technologies (PET), Privacy Management Systems (PMS) and Identity Management systems (IMS), the Drivers thereof and the need for standardization. CEN Workshop Agreement", CEN, Brussels. ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/DPP/CWA15263-00-2005-Apr.pdf

Gal, R., and M. van Genuchten (1996), "Release the Embedded Software; the Electronics Industry in Transition," *International Journal of Technology Management*, June.

Gartner (2007a), "Gartner Says Service Providers Must Prepare Now for the Software as a Service Wave", Gartner, Stamford, available at: www.gartner.com/it/page.jsp?id=501991 (last accessed 29.09.2008).

Gartner (2007b), "Gartner Says Worldwide Software as a Service Revenue in the Enterprise Application Software Markets to Grow 21 Percent in 2007", Gartner, Stamford, www.gartner.com/it/page.jsp?id=511899 (last accessed 29.09.2008).

Gartner (2007c), "Gartner Says Worldwide Security Software Revenue Will Reach $9.1 Billion in 2007", Gartner, Stamford, www.gartner.com/it/page.jsp?id=500694 (last accessed 29.09.2008).

Gasser, U. and J. Palfrey, (2007), "Breaking Down Digital Barriers: When and How ICT Interoperability Drives Innovation", Berkman Publication Series, Available at: http://cyber.law.harvard.edu/interop/downloads.html.

Gulliksen, J., and Harker, S. (2004), "The software accessibility of human-computer interfaces – ISO Technical Specification 16071", Universal Access in the Information Society, Vol. 3.

IDC (2007), *IDC Says Expanding Needs for Mobile Device Security Software Will Drive Revenue Growth Over the Next Five Years*, Press Release, http://www.idc.com/getdoc.jsp?containerId=prUS20633707.

IEEE (2004), Guide to the Software Engineering Body of Knowledge (SWEBOK), executive eds.: Alain Abran and James W. Moore, eds.: Pierre Bourque and Robert Dupuis, Los Alamitos, CA, USA. (SWEBOK is also published as ISO/IEC TR 19759:2005.)

IFIP (1988), "The Charter of WG 10.4 on Dependable Computing And Fault Tolerance" (established 1980, revised 1988), the International Federation For Information Processing, http://www.dependability.org/, accessed on 9 July 2008

Information Commissionner's Office (ICO) (2007), « Data Protection Guidance Note : Privacy Enhancing Technologies », ICO, Wilmslow, United Kingdom. www.ico.gov.uk/upload/documents/library/data_protection/detailed_specialist_guides/privacy_enhancing_technologies_v2.pdf

Instituto Nacional de Tecnologías de la Comunicación (Inteco) (2008a), «Studies and Reports », León, Spain. www.inteco.es/Security/Observatory/Publications/Studies_and_Reports/

Instituto Nacional de Tecnologías de la Comunicación (Inteco) (2008b), « Study on security incidents and needs in Spanish small and medium-sized enterprises», León, Spain, www.inteco.es/Security/Observatory/Publications/Studies_and_Reports/estudio_pymes_29.

ISO (2003), *Proposed Draft Technical Report for: Information technology -- Learning, education, and training -- Management and delivery -- Specification and use of extensions and profiles*, International Organisation for Standardization, available at: http://old.jtc1sc36.org/doc/36N0646.pdf.

ITEA (2005), "Software intensive systems in the future", Information Technology for European Advancement, available here: http://www.itea-office.org/publications, last accessed 08.08.2008.

Ito, S. (2007), "Developing global society through innovation in the software sector", presentation at the OECD Conference on Innovation in The Software Sector, Caceres, Spain, 30 November 2007.

Kavcic, A. (2005) Software Accessibility: Recommendation and Guidelines, EUROCON 2005, available here: http://lgm.fri.uni-lj.si/alenka/web/Eurocon2005.pdf .

Lueders, H. (2005), "Interoperability and Open Standards for eGovernment Services", Initiative for Software Choice (ISC), Brussels.

Lyu, M.R. (2007), "Software Reliability Engineering: A Roadmap", International Conference on Software Engineering, 2007 Future of Software Engineering.

MacWilliams, A. (2004), "Software Development Challenges for Ubiquitous Augmented Reality", Technische University Munchen, Germany.

Ministry of the Interior and Kingdom Relations, the Netherlands (2004), "Privacy enhancing Technologies: White Paper for Decision Makers", www.dutchdpa.nl/downloads_overig/PET_whitebook.pdf.

METI (2007), *Interoperability Framework for Information Systems Version 1*, Ministry of Economy, Trade and Industry, Japan, June.

Mooney J.D. (1997), "Bringing Portability to the Software Process", Technical Report TR 97-1, West Virginia University, Dept. of Statistics and Comp.Sci., 1997.

Niemelä, E., and Latvakoski, J. (2004), "Survey of Requirements and Solutions for Ubiquitous Software", Mobile Ubiquitous Computing Conference 2004, Washington DC, USA.

Nikkei (2003), "Informatization State Survey", Nikkei Computer (magazine in Japanese), as cited by the Ministry of Economy, Trade and Industry, http://www.nikkeibp.com/html/pub/computer.html.

OECD (1980), « OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data », OECD, Paris.

OECD (2002), "Guidelines for the Security of Information Systems and Networks: Towards a Culture of Security", OECD, Paris.

OECD (2004), Digital Broadband Content: Mobile Content, New Content for New Platforms, OECD 2004, Paris.

OECD (2004a), "Summary of Responses to the Survey on the Implementation of the OECD Guidelines for the Security of Information Systems and Networks: Towards a Culture of Security », OECD, Paris.

OECD (2004b), « Biometric-Based technologies », OECD, Paris.

OECD (2005a), "The promotion of a Culture of Security in OECD Countries », OECD, Paris.

OECD (2005b), « Scoping Study for the Measurement of Trust in the Online Environment », OECD, Paris.

OECD (2006a), *Information Technology Outlook*, 2006, OECD, Paris.

OECD (2006b), *Mobile Commerce*, OECD, Paris.

OECD (2007), "Participative Web and User-created Content Web 2.0, Wikis and Social Networking", Sacha Wunsch-Vincent and Graham Vickery, available here (last accessed 2 August 2008): http://213.253.134.43/oecd/pdfs/browseit/9307031E.PDF.

OECD (2008a), « The development of policies for the protection of critical information infrastructures (CII) », DSTI/ICCP/REG(2007)20/FINAL, OECD, Paris.

OECD (2008b), *OECD Recommendation on Policies for the Protection of Critical Information Infrastructures*, OECD, Paris.

OECD (2008c), *Malicious Software (Malware): A Security Threat to the Internet Economy*, DSTI/ICCP/REG(2007)5/FINAL, OECD, Paris.

OECD (2008d), *Scoping Paper on Online Identity Theft*, DSTI/CPP(2007)3/FINAL, OECD, Paris

OECD (2008e), *Guidance on Radio-Frequency Identification*, OECD, Paris.

OECD (2008f), *Radio-Frequency Identification: a Focus on Security and Privacy*, OECD, Paris.

OECD (2008g), *Shaping Policies for the Future of the Internet Economy*, OECD, Paris.

OHA (2008), *Open Handset Alliance*, available at: http://www.openhandsetalliance.com/, (last accessed 07.08.2008).

Pan, Jiantao (1999), "Software Reliability", Carnegie Mellon University, available at: http://www.ece.cmu.edu/~koopman/des_s99/sw_reliability/#reference (as of 14 July 2008).

Parker, G. and Van Alstyne, M. W. (2008), *Innovation, Openness, and Platform Control*, July 24, available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1079712 (last accessed 08.08.2008).

Pfleeger, C., Pfleeger, S. (2007), « Security in Computing », Prentice Hall.

Reed P.S., Gardner-Bonneau, D., and Isensee, S. (2004), "Software Accessibility Standards and Guidelines: Progress, Current Status, and Future Development", Universal Access in the Information Society, Vol.3.

Roman G. et. al (2000), "Software Engineering for Mobility: A Roadmap", International Conference on Software Engineering, Proceedings of the Conference on The Future of Software Engineering, 2000.

Roman, G., Picco, G.P. and Murhphy, A.L. (2000), "Software Engineering for Mobility: A Roadmap", International Conference on Software Engineering, Ireland.

Software and Information Industry Association (2001), « Software as a Service: Strategic Backgrounder », Software and Information Industry Association, Washington.

Sousa, J.P., and Garlan, D. (2002), "Auro: an Architectural Framework for User Mobility in Ubiquitous Computing Environments", 3rd Working IEEE/IFIP Conference on Software Architecture, Montreal, Canada.

TEITAC (2008), Report to the Access Board: Refreshed Accessibility Standards and Guidelines in Telecommunications and Electronic and Information Technology, April, available here: http://www.access-board.gov/sec508/refresh/report/#1.

Tsilas, N. L. (2007), "Enabling Open Innovation and Interoperability: Recommendations for Policy-Makers", *ACM International Conference Proceeding Series*, Vol. 232, 1st International Conference on Theory and Practice of Electronic Governance, Macao, China.

UK Cabinet Office (2008), "Data Handling Procedures in Government: Final Report", www.cabinetoffice.gov.uk/~/media/assets/www.cabinetoffice.gov.uk/csia/dhr/dhr080625%20pdf.ashx.

Van Eeten, M., Bauer, J., De Bruijne, M., Groenewegen, J., Lemstra, W. (2008, forthcoming), "Economics of Malware. Security Decisions, Incentives and Externalities", STI Working Paper, OECD, Paris.

van Genuchten, M. (2007), "The impact of software on the electronics industry", *IEEE Computer*, January.

van Genuchten, M. (2008), *The Impact of Software Growth on Companies, Industries and National Economies*, issues paper prepared for the meeting of the OECD Advisory Expert Group on Innovation in the Software Sector, Tokyo, 7 October 2008.

WSJ (2005), "Shakeout Could Jolt Electronics," Wall Street Journal, 27 January.

Weiler, J.H.H. (2005), "The Legal Regulation of Software Interoperability in the EU", Jean Monnet Working Paper 07/05, New York.

Yoshida, M. (2006), Japan Perspectives on ICT Accessibility, presentation at CSUN 21st Annual International Conference.