

**Proceedings of  
Umeå's 11<sup>th</sup> Student Conference in  
Computing Science  
USCCS'07**

*Edited by  
Jürgen Börstler, Håkan Gulliksson and Lars Erik Janlert*

**UMINF 07.08  
ISSN-0348-0542**

UMEÅ UNIVERSITY  
Department of Computing Science  
SE-901 87 Umeå, SWEDEN



## Preface

Umeå's Student Conference in Computing Science is the highlight of a "conference course" in our Computing Science curriculum. The objective of this course is to give students a forum, where they can actively participate in scientific research and development. The conference format was chosen to provide a realistic environment for the presentation of their research results.

The "conference course" is a practical course for students interested in research and introduces them to

- independently researching an interesting topic;
- using a foreign language (English);
- writing scientific reports on their work;
- presenting their work at a conference.

This year was the eleventh offering of the course with a total of 43 registered students. Of these 43 students, 39 actively participated in some part of the course and 23 of those eventually submitted a full paper. Of the 23 submissions, we accepted 18 for presentation at the conference and publication in the proceedings. In addition to the 18 papers from this course, we also included in the proceedings one late paper from last year's course. Due to the tight submission, review, print cycle, this paper did not make it into last year's proceedings.

Each submission received at least two independent reviews. We would like to thank all reviewers who helped to review all papers within a very short time frame.

Please check the conference course home page for further information on the course (<http://www.cs.umu.se/kurser/TDBD18/>).

Umeå, May 2007

Jürgen Börstler  
Program Chair  
USCCS'07

IV

### **Program Committee**

Jürgen Börstler (Program Chair)

Håkan Gulliksson

Lars Erik Janlert

### **Additional Reviewers**

Jerry Eriksson

Claude Lacoursière



# Table of Contents

<b>Preface</b> .....	III
<b>Social Aspects of Computing</b>	
Augmenting Implants in Humans—An Overview .....	1
<i>Per Eriksson</i>	
Translation—more than just words .....	11
<i>Hanna Ojansivu</i>	
Social Interaction in Virtual Worlds .....	19
<i>Göran Lundin</i>	
<b>Miscellaneous</b>	
The relevance of aesthetics to the success of the Apple iPod .....	31
<i>Mathias Bergmark</i>	
Teaching Project Management Using Computer-Based Learning Tools ...	43
<i>Jonas Bergström</i>	
The Benefits and Limitations of Self Organizing Feature Map Clustering .	53
<i>John Edwards</i>	
Performance Characteristics of String and List Classes in Java 1.6 .....	69
<i>Timo Elverkemper</i>	
<b>Interaction Design</b>	
Designing for Real-Time Collaboration Through Small Screens .....	81
<i>Reza Assareh</i>	
Designing emergent interaction: Adaptive interfaces with emergent behaviours .....	97
<i>Jakop Berg</i>	
Mobile phone interfaces for the visually impaired—A study .....	109
<i>Fredrik Björnskiöld</i>	
<b>Internet / Wireless</b>	
Evaluation of Quality of Service Performance in Wireless Local Area Networks .....	121
<i>Muhammad Shahid Manzoor</i>	

Autonomous Peers Collaboration.....	133
<i>Davide Neri</i>	
Textual Advertisement Models—A Comparative Look .....	153
<i>Abubakr Saeed</i>	
Performance Evaluation of Slow Contention Window Schemes for Wireless Local Area Networks .....	163
<i>Imran Siddique</i>	
Frameworks for Context Aware Ad Hoc Communication Systems—A Survey .....	173
<i>Khurram Ali Khan</i>	
<b>Management</b>	
A multilayered decision model for command and control systems .....	183
<i>Mikolaj Kunc</i>	
Earned Value Management as Project Follow up .....	197
<i>Stefan Lindkvist</i>	
An investment perspective on usability .....	209
<i>Anders Moberg</i>	

# Augmenting Implants in Humans

## – An Overview

Per Eriksson

Student Conference in Computing Science  
Umeå University, Sweden  
dit03pen@cs.umu.se

**Abstract.** This paper deals with the concept of the cyborg, the union of man and machine, and the parts that make up a cyborg. Different types of cyborgs are discussed in an effort to determine what they represent and when, if ever, they might become a reality. To give a better understanding of how far scientists have come some examples of cybernetic hardware will be discussed, some in use today, some still only as ideas. Special attention will be given to RFID and GPS technology which could prove beneficial to have implanted in your body at all times. RFID-tags could provide excellent access and security in identification and GPS is a very sophisticated tool for navigation that could be helpful in everyday life as well as professionally. Some thought is given to the dangers of cybernetic implants, not only from a medical standpoint, but also from a privacy perspective and the implications in social structures. Lastly, I try to sum everything up to answer the question whether cyborg bodies are something to strive for, or if its simply too little gain and too big a risk.

## 1 Introduction

In the latest Bond movie, *Casino Royal*, Mr. Bond gets a microchip implanted in his forearm, using a large syringe, so that the MI5 can track him wherever he goes. Seems far fetched, doesn't it? But the technology is not as far off as you might think. The miniaturization of computers and the growing number of worldwide communications systems are allowing for more and more advanced tasks to be performed by smaller and smaller devices.

As we surround ourselves more and more with computers it makes sense to invent ways to help us communicate with them more effectively, and through them with other people or information. The methods we use today for communicating with our technology are quite diverse; there is the traditional keyboard and mouse system, and different types of information readers such as a credit card reader or a retinal scan camera. But what if we could control our computers by simply thinking? How compatible are man and machine really?

With this paper I seek to answer some questions about implant technology and its implications for our society. Namely what types of implants are represented in science fiction, and are any of them likely to be realized in the

foreseeable future? I will also try to give some arguments concerning the benefits and deficits of putting technology under our skin instead of simply carrying it. Furthermore I will look at some examples of technological implants that are already in use and some which are still some time away.

### 1.1 Cyborgs

A cyborg (short for: cybernetic organism [1]) is the union of living organic matter and cybernetic hardware working together as one being or system. In the present paper the living matter will be human beings. Cybernetic hardware [2] is mechanical parts with sensors and effectors controlled by rules using feedback loops and algorithms for behaviour, in most cases a computer controlled mechanical apparatus. To understand the distinction from other types of hardware consider a one legged man. The use of a cane would permit the man to walk, but it would not make him a cyborg. A modern leg prosthesis on the other hand could be connected to his central nervous system and interpret the signals from key nerves as instructions as to how to react. The man would be able to walk and could be considered a cyborg. This, however, is not how most people envision cyborgs. The common view of a cyborg is a human where healthy tissue has been removed and replaced by superior mechanical parts. The founders of the word cyborg, M. Clynes and N. Kline, stated that it is a “self-regulating man-machine hybrid” [1]. This means that our one-legged man from the example above is, in fact, not a cyborg since his prosthesis would require regular maintenance and have its batteries recharged. Therefore he would not be “self-regulating” [1].

**Classifications** Writers have envisioned cyborgs in many ways through history and now some of their fantasies are coming true. In his paper “From Cyborg Fiction to Medical Reality” Craig M. Klugman describes four different types of cyborgs from literature and movies. These types of cyborgs are: the transplantable body, the disembodied mind, the super body and the linked body [3], described in more detail below.

The “transplantable body”-type cyborgs are humans who have suffered some kind of injury and have replacement parts that replicate the function the human had before the injury as closely as possible. Again thinking about the “self-regulating” qualities of the definition of a cyborg only some of today’s prosthetics would constitute a “transplantable body”-type cyborg. Some of these implants are: pacemakers, mechanical hearts and Cochlear implants which are connected to the nerves in the ear and give a form of hearing to otherwise deaf patients.

For the next type of cyborg portrayed in fiction one must understand the Cartesian notion of the body and the mind. The 17th-century philosopher René Descartes was convinced that the body and mind were two different entities, and could continue to live even when separated. This is the basis for the “disembodied mind”-type cyborg. The idea is that the human mind could be downloaded onto a computer and still be uniquely identifiable as the same mind. A “disembodied mind”-type cyborg is a human who has suffered injuries too great to be

repaired so the mind is instead placed inside a wholly mechanical body with a mechanical brain. It is important to point out that philosophers worldwide have been debating the Cartesian notion (or dualism) since Descartes' death in 1650, and there is no consensus among them [4].

The "super body"-type cyborg is close to the "transplantable body"-type with the difference being that it seeks to improve on the human's functionality. Klugman claims that some cosmetic surgery and the use of steroids could be seen as examples of humans using technology to gain superhuman attributes.

The last type, the "linked body"-type is a human whose only mechanical parts are for allowing for communication with other humans with the same implants. The implants would allow a mind to travel the network in a Cartesian way, and the body would be a tool for the different minds to affect the physical world.

## 1.2 Information Age

Some of Klugman's cyborgs types are too visionary to say much about, but the "super body"-type is coming true in different labs and hospitals around the world today. We may not have the technology to create superhuman limbs yet but the prosthesis industry is catching up with evolution very fast. What we do have today, that far supersedes human capabilities, are vast communication networks spanning the globe. Using the telephone net or internet allows us to gather information and communicate instantaneous over distances that made such things impossible before today. We are now in what some refer to as the "information age" [5] where information is of more value than any other currency or goods [2]. The information is, in fact, the currency in some cases. The stock market, for example, is dictated by the trends and expectations of company profits. Should someone know the sales of a large company before everyone else, he or she would be able to make a lot of money through selling, or buying, stocks in that company. Hence the information has real value. This is true for a lot of types of information, and the need to be constantly updated is increasing as people realize this.

Our way of living has spawned a new ideology called posthumanism. This is a philosophy that embraces science, technology, and information to the extent that it becomes the centre of existence. Hayles [5] and Bendle [6] explain that posthumanists believe in the Cartesian notion, discussed above, and that the existence of life is only ever repeating information patterns that can be seen everywhere, not only in life. The posthuman philosophy encourages the development of cyborgs as they believe that the body is just the first prosthesis our minds learn to control, and other, better, devices could be mastered as well. Bendle claims that there are a few groups and organizations that call themselves posthumanists, but that they are in most cases very naïve in their approach to science [6].

## 2 Cybernetic Implants

There are some implants that can be considered cybernetic hardware today but they are mostly of the “transplantable body”-type. In this chapter I will first examine two types of implants that are quite realistic in detail; the RFID tag implant, and the GPS transceiver implant. Then I will present a few examples of research projects that deal with different types of implants that might help us in the future.

### 2.1 RFID implants

Having a microchip implanted in your body is already becoming commonplace. There are several companies that provide customers with this technology for different reasons. These types of microchips contain data which can be accessed from a small distance using radio signals and they do not need a power source to function, neither are they in any way connected to the body, except for being implanted under the skin [7, 8]. They can be thought of as bar codes; in fact some companies, such as Wal-Mart [9], are beginning to use them as that since they are very cheap when ordered in large numbers.

**Advantages.** The first thing that comes to mind when thinking about RFID implants is the possibility to quickly, positively identify people. This method of identification has been used in pets for years and is considered safe for humans. The idea is that the chip should contain a unique number linking to a database of all people currently using the system. This is beneficial to everyone as it could be used in airports, shops and clubs and also for identifying corpses or people otherwise unable to communicate their identity. This is often called ULI (universal lifetime identification).

In the Baja Beach Club, in Holland, VIP customers may have a chip implanted in their arms and receive a lot of advantages in the club such as paying in the bars by a wave of the hand [10]. Another example is an American company that offers customers up to 14 individual RFID chips implanted in different body parts to be used for identifying a corpse should the user be in a terrorist attack [11].

The advantage of these kinds of systems is that you cannot misplace your identification and should it become an international system you would no longer need to carry a passport or other types of identification. The downside is that the chip is useless unless the reader has access to the updated database containing the information.

**Dangers.** The biggest risk with ULI is the threat to people’s integrity. Being identifiable at all times can be troublesome and uncomfortable. It is anyone’s right to be anonymous at any time they wish to be, so a way to turn your ULI off temporarily might be a good idea.

Another risk is that of forgery, which in the case of ULI means identity theft. If the RFID tag data is accessible to anyone who gets close with a RFID reader someone might copy the information and implant a cloned ULI for the purpose of disguising him- or herself. These things happen today with people pretending to be someone else when they buy cars or goods via internet. The good thing is that identity theft will be a lot harder if and when ULI is in use than it is now. Besides cloning the tag, the only other way to steal someone's identity is to actually removing the tag from inside the body, which is not very easy to do without being noticed. There are good ways to prevent cloning of ULI's in encryption techniques.

The risk of the RFID-tag actually malfunctioning is very low, but the small memory in the tag is not only readable, but also writeable. This could cause problems as the destruction of information is very popular, and hackers and other people might benefit from changing or erasing someone's ULI. This can also be made difficult by encrypting the information sent by tags and readers.

**Technology.** The RFID system consists of three parts: the antenna, the transceiver with a decoder and the transponder (or RFID-tag). The antenna and the transceiver make up the reader and the transponder is what is put on or in something. The reader provides the energy the RFID-tag needs to communicate and therefore it needs no batteries and can remain usable for decades without maintenance (passive RFID). The range of a passive RFID-tag depends on the reader and the surroundings and is usually no longer than one meter and sometimes it is as short as a few centimetres. The amount of data that can be stored in a passive RFID-tag is usually no more than 128 bytes. This data is readable and writeable using the proper reader. A passive RFID-tag can be made very small; a common estimate is about the size of a grain of rice [7,8].

**Medical issues.** The insertion of a RFID transmitter is done using a large syringe. The transponder is placed in the sub dermal skin where it remains indefinitely. The procedure is at most uncomfortable and the risk of complications is low. The technology has been in use on pets for several years and there have been some cases where the transponder has wandered in the pet's body but this is easily remedied by creating a biocompatible glass casing for the chip which makes the organic tissue grow attached to the glass preventing it from moving inside the body [7].

## 2.2 GPS implants

A GPS transceiver is a small device that registers radio wave signals from satellites orbiting the earth. This allows the GPS transceiver to calculate its position in three dimensions with a margin of error of approximately one meter [12]. A GPS transceiver is not very big. In fact an Australian company recently showcased their new model which is about the size of a baby fingernail [13]. Implanting such a chip in a human being would not be very useful though. To be able to get

something out of the GPS transceiver you need to present the data somehow. This can be done in many ways; presenting the coordinates on a screen with a corresponding map or sending the coordinates to a server using the telephone net. Both of these methods require gadgets that are much bigger than the actual GPS transceiver. Also a GPS transceiver, and some type of communications device, both require a power source. Since the device would be positioned inside the abdomen, body heat might be a viable power source in the future.

**Advantages** GPS is primarily used for navigation. Early on the GPS transponders were less accurate which made navigation indoors or near tall buildings somewhat unreliable, but today the technology has improved and as long as you stay above ground GPS transponders are likely to be able to give you your exact location. When out in the wild or perhaps on vacation in an unfamiliar city, always knowing your exact position is a great advantage, but is it really important enough to put a computer chip in your body? Most people don't feel the need to know their exact coordinates in their everyday life as they usually find their way to work or to the store anyway, and there are portable GPS-units available at a reasonable price. The real advantages come when you are able to see where everyone else is in relation to you and to the world. You could find your blind date in seconds, you could avoid your boss all day, or you could see at what club in Thailand your daughter is spending her holiday.

There are also all the common uses of a portable GPS-unit as well. Some models have functionality that aids you when exercising by tracking your runs and calculating different things like mean speed. Some GPS's use interactive maps where people, using the Internet, can mark special locations so that other users can find them and experience them themselves. Some people even use their GPS's to "draw" paintings by walking around in odd shapes.

**Dangers** The dangers to the person (or cyborg), except for medical dangers, in this case depend very much on how the communication with the GPS transponder would be carried out. If, for example, the person's coordinates would constantly be sent to a web server for any purpose there is the danger of the data ending up in the wrong hands. However secure the system; one can never be 100% safe from leaking data or human error. To give the users the privacy they require, the possibility to turn the device on and off at any given moment is desirable. Since acquiring your coordinates using GPS does not give any kind of information to anyone, you could still see you own, and others, coordinates even though you would not be visible to others.

**Technology** There are at least 24 operational GPS satellites orbiting the earth at all times. Each of these satellites carries a very accurate atomic clock on-board. Each satellite transmits radio wave signal containing information about the satellites current position and the exact time of transmission. A GPS transponder listens to these radio signals and when the transponder receives signals from at



least four satellites it is able to calculate its position in three dimensions. The GPS technology we use today became fully functional 1995 [12].

The actual GPS transponder is small and it requires little power, and the update frequency can be lowered to further decrease the use of power. To communicate with a server an implant would need to be able to wirelessly access the mobile phone net or the internet from anywhere in the world. Currently the only technology that can stand up to these demands is Iridium mobile phone technology. Iridium is a system of satellites, much like the GPS system, that orbit the earth and provides communication with the telephone net in all parts of the world. Iridium does not work very well when indoors or in dense vegetation [14].

Batteries to power both the GPS transponder and an Iridium communications device for a long time, preferably over 10 years, would be too big to be implanted in a human. There is some research done in the field of fuel cells however, which indicates small cells providing performance many thousands of times greater than batteries today. Should these kinds of fuel cells become a reality, the GPS implant might too.

**Medical issues** Implanting a GPS transponder, a battery, and a device for communicating with something outside the body or the nervous system is a big risk to the patient. The only place to put something of that size is the abdomen. This is done when implanting mechanical hearts, as the battery pack needed to provide the heart with power is quite large. These kinds of batteries last up to 10 years before having to be replaced or recharged [15].

Although researchers do not agree some claim that the exposure to the electromagnetic fields created by mobile phones are hazardous to human tissue and most of all the brain. This leads to think that implanting a similar device in a human is not to be taken lightly.

### 2.3 Implants Today

Around the world scientists and doctors are making great progress in understanding the human body and mind as well as creating more and more advanced technology. Today, very advanced prosthetics exist, that closely mimics the functions of natural, healthy limbs and can be controlled by the patient by simply thinking. One such prosthesis is the C-leg, which is a modern hydraulic leg prosthesis that allows above-knee amputees to run and use stairs as if they were not disabled. The prosthesis uses different kinds of sensors and a microprocessor to determine the desired motion and then the hydraulic system puts the changes into effect. [16]

Another, even more impressive, prosthesis is the Dobbelle Artificial Vision System (DAVS), giving eyesight to otherwise blind patients. DAVS is a low resolution video camera that is mounted to a pair of glasses. The output of each pixel is sent as electrical impulses directly to selected parts of the patient's visual cortex, and is interpreted by the brain as images. There are numerous patients

who are currently using DAVS, and some of them have had vision earlier in life, but lost it due to illness or accidents. [17] One of these patients describes the vision gained as very crude but it still gives an immense feeling of freedom. [18]

Nicolelis has been testing brain-machine interfaces on monkeys, and has been very successful [19]. One of his most well known experiments was getting a monkey to move a pointer on a computer screen by simply thinking. The monkey had sensors connected to nerves in his brain, which regulate movement of the arm, and was taught to move a joystick controlling the pointer over the screen. After a long time of calibrating and testing the pointer was programmed to use the signals from the sensors in the monkey's brain, rather than the signals from the joystick. This eventually allowed the monkey to realize that it had only to think about moving its arm in order to move the pointer. This ability to control apparatus by thinking is presumed to be possible for humans as well, and it has enormous implications on our technology and life in general.

#### **2.4 Future Implants**

With these kinds of implants already available it is hard to tell what the future might hold, but initially we will probably see improvements on the types of implants we see today. Should the performance of an artificial limb in some way become greater it is possible that people will want to remove healthy tissue and/or limbs to have it replaced with mechanical parts.

The types of implants that are probably further away are implants that add extra functionality, for example having your cell phone connected to your brain in order to communicate without actually speaking. Another example is artificial wings giving the ability to fly. Yves Rossy from Switzerland has created wings that let him fly like a jet-plane for up to six minutes [20]. These wings are cumbersome and the capabilities are very limited. With the wings Rossy use today he is not able to take off on his own, but must be towed by an aircraft.

The work of Nicolelis is creating a very interesting future where controlling our technology might become easier than ever before [19]. The brain-machine interfaces he has created only allow for scanning the brain, no feedback can be given yet. But combined with variations of DAVS the possibilities seem endless [17].

### **3 Conclusion**

Based on the examples I have discussed in this paper I have to say that science fiction is sometimes eerily accurate. Most of the technology in these examples has been developed in the last decade, and yet many writers envisioned similar devices some fifty years ago and earlier [3]. One thing that science fiction has done is make the vision of the cyborg something intimidating and alien, but as this paper has proven that is not the case. A cyborg can be merely a human with a pacemaker, and contrary to what many may think cyborgs are already among us.

I have touched on privacy issues with some implants, and that is an area of great concern, but something I have yet to reflect upon is the fear of discrimination. Some people will see the development of new “super body”-type cybernetic limbs and other parts as a threat to social structures. If, for example, a logistics company would give all their employees new bodies, twice as strong as normal humans, to prevent work related injuries and accidents, they would later on be more qualified for other jobs simply because of their mechanical parts. This is true for much simpler implants as well, and could in a worst case scenario lead to elitism and segregation. I, however, feel that this would be a form of economical segregation, and we already have that. In wealthy countries people have all sorts of technological tools which give them an edge on the people that don't. The problem lies not in putting these tools under our skin.

To implant, or not to implant? To me it seems a question of personal principals rather than a simple yes or no. The benefits of having a GPS transceiver in your abdomen instead of carrying one in your cell phone are that you cannot misplace it. In abstract this is the only advantage. The drawbacks are the surgical incision, which is always a risk, and the risk of complications with having non organic materials inside your body. But imagine having the ability to instinctively know your way anywhere in the world. Combining the technologies I have presented might make this possible in our lifetime.

## References

1. Clynes, M.E., Kline, N.S.: *Cyborgs and Space*. *Astronautics* (1960) Reprinted in *The Cyborg Handbook*, ed. Chris Hables Gray, with Heidi J. Figueroa-Sarriera and Steven Mentor (New York: Routledge, 1995), 30-31.
2. Wiener, N.: *The Human Use of Human Beings: Cybernetics and Society*. Da Capo Press, Cambridge, Massachusetts (1988)
3. Klugman, C.M.: From cyborg fiction to medical reality. *Literature and medicine* **20**(1) (2001) 39–54
4. wikipedia.org: René descartes (2007) [http://en.wikipedia.org/wiki/René\\_Descartes](http://en.wikipedia.org/wiki/René_Descartes), accessed 2007-05-07.
5. Hayles, N.K.: *How We Became Posthuman: Virtual bodies in cybernetics, literature, and informatics*. University of Chicago Press, University of Chicago (1999)
6. Bendle, M.F.: Teleportation, cyborgs and the posthuman ideology. *Social Semiotics* **12**(1) (2002) 45–62
7. technovelgy.com: What is rfid? (2007) <http://www.technovelgy.com/ct/Technology-Article.asp?ArtNum=1>, accessed 2007-05-07.
8. autoid.org: Active and passive rfid (2002) [http://www.autoid.org/2002\\_documents/sc31\\_wg4/docs\\_501-520/520\\_18000-7\\_whitepaper.pdf](http://www.autoid.org/2002_documents/sc31_wg4/docs_501-520/520_18000-7_whitepaper.pdf), accessed 2007-05-07.
9. walmartstores.com: Radio frequency identification (2007) <http://walmartstores.com/GlobalWMStoresWeb/navigate.do?catg=339&contId=6181>, accessed 2007-05-07.
10. technology.guardian.co.uk: I've got you under my skin (2007) <http://technology.guardian.co.uk/online/story/0,3605,1234827,00.html>, accessed 2007-05-07.
11. limbidsystem.com: Limb identification system (2007) <http://limbidsystem.com/>, accessed 2007-05-07.

12. National Air and Space Museum: Gps: A new constellation (2007) <http://www.nasm.si.edu/gps/index.htm>, accessed 2007-05-07.
13. rakon.com: Rakon develops world's smallest receiver (2006) [http://www.rakon.com/whatsnew/display?article\\_id=-80&template=news](http://www.rakon.com/whatsnew/display?article_id=-80&template=news), accessed 2007-05-07.
14. iridium.com: Rakon develops world's smallest receiver (2007) <http://www.nasm.si.edu/gps/index.htm>, accessed 2007-05-07.
15. Rakobowchuk, P.: Man with no pulse considered a medical breakthrough (2006) <http://www.theglobeandmail.com/servlet/story/RTGAM.20061213.wheart1213/BNSStory/specialScienceandHealth/home>, accessed 2007-05-07.
16. Otto Bock Health Care: C-leg microprocessor knee (2007) [http://www.ottobockus.com/products/lower\\_limb\\_prosthetics/c-legproduct.asp](http://www.ottobockus.com/products/lower_limb_prosthetics/c-legproduct.asp), accessed 2007-05-07.
17. Dobelle, W.H.: Artificial vision for the blind by connecting a video camera to the visual cortex. *ASAIO journal* **46** (2000) 3–9
18. CBS News: Technology brings sight to the blind (2002) <http://www.cbsnews.com/stories/2002/06/13/earlyshow/health/main512160.shtml>, accessed 2007-05-07.
19. Nicolelis, M.A.L.: Computing with neural ensembles (2007) <http://www.neuro.duke.edu/faculty/nicolelis/index.html>, accessed 2007-05-22.
20. Rossy, Y.: Fusion man (2007) <http://www.jet-man.com/prod/index.html>, accessed 2007-05-07.

# Translation – more than just words

Hanna Ojansivu

Department of Computing Science  
Umeå University, Sweden  
dit03hou@cs.umu.se

**Abstract.** Translating only the words is not always enough for a software product or website to be accepted in a different country. Colours can have the opposite symbolic meaning from that in the original country and the behaviour of people may differ. Those who take care to adapt their products to their users will more likely be successful. Enough space must be available for the translated text to expand. Standards must be adhered to. Culturally specific items must be isolated so they can easily be replaced. What symbols to insert instead can be found by auditing other local successful websites to see what is commonly used. Anthropology can be used to find values, attitudes and behaviour specific to a culture. Known cultural traits can be used to predict a suitable design. To discuss around, a visualization of the cultural characteristics can be made. Even though the web and globalization in general spreads ideas like never before, there is too much that differs between cultures to ignore.

## 1 Introduction

When planning to introduce a software product in another country, it is easy to make the mistake of thinking that all that needs to be changed is the interface language. But it is not only the language that differs between different countries and cultures. The way people think, interpret or do things can also be different. These differences between cultures should be reflected when computer programs, websites or other technology are translated to other languages. Simply replacing the words is not necessarily enough to make the users in a foreign culture feel comfortable using the product. Symbols such as a picture or icon, colour combinations and the structure of how things are done should also mirror cultural differences. Depending on the application, infrastructure, speed of internet connections or availability and reliability of other services might also be a relevant concern.

Below are motivations to why more thorough studies and adaptations need to be done before launching a product in a foreign country. After that, methods and suggestions for how one can work with these issues are discussed.

## 2 Why Bother?

As the world goes towards globalization the great potential of the world's markets may seem tempting, but products or services made in a different context might

not at all become a success in the new country. For example, a study [1] about the possibility of widespread e-commerce in China found that there are a few factors that could slow down the process of widespread use. One of them is the issue of trust. The history of China has taught its people that the thing to trust is a strong individual relationship. There is also a strong socialization effect while doing business. This is hard to accomplish in a traditional e-commerce site. Another factor is the common Chinese practice of negotiating about the price which is also not possible on many e-commerce sites.

If a product or service is going to be used a lot, usability is an important issue and one of the aspects to address is the diversity of users, including (but not limited to) culture. Those who manage to tune their products to local needs are likely to be more successful [2]. Another advantage is that if preparations are done for customizing for one purpose, it is then easier to make other changes too to allow for different preferences, hardware etc.:

“Evidence is accumulating that designs that facilitate multiple natural-language versions of a website also make it easy to accommodate end-user customization, convert to wireless applications, support disabled users and speed modifications. The good news is that satisfying these multiple requirements also produces interfaces that are better for all users. Diversity promotes quality.” [3]

As computer products become increasingly integrated with our lives it becomes more important that they are adapted to our culture. It is not possible to ask everyone to adapt to the product [4]. The products must adapt to the culture.

Examples of differences between cultures are the way people respond to and use space (e.g. in conversational distances, in office layout), treat time (e.g. in assumptions about what an appointment for 2pm or dinner at 8 actually means), and interpret signs in their environment [4]. If employees can work individually or must have a team [5], the relation between employees and managers [6, 5], the view of the customer [7] and attitudes towards various other phenomena, for example credit cards [7, 1] also varies between cultures. Even the basic user requirements may differ between cultures, not only surface level adaptation of interface elements [8]. An example of this is the Swedish ATMs’ big buttons. They can be used even while you’re wearing thick gloves needed cold winter days [9]. For some examples of how the symbolic of colours differ quite radically between cultures, see Table 1 [10].

The perception of what kind of persons the intended users are can have a great influence on the likelihood of adoption of a technology. In societies where people’s lives are strongly regulated by social status this is a critical issue [8]. Since the common knowledge, values and attitudes shape people’s interaction with their environment, culture has to be regarded as a powerful variable affecting users’ expectations and behavioural possibilities. Thus culture can determine people’s response to a machine, including misuse or no use at all. An example is the introduction of ATMs in Mumbai, India [8]. India is a collective culture where members of a group help each other. This has the effect that people are used to borrowing cash from their social network and thus the need for ATMs

**Table 1.** Example colour associations for some cultures (Used with permission from the Association for Computing Machinery ©1993 ACM 0-89791-575-5/93/0004/0342...\$1.50).

Culture	Red	Blue	Green	Yellow	White
United States	Danger	Masculinity	Safety	Cowardice	Purity
France	Aristocracy	Freedom Peace	Criminality	Temporary	Neutrality
Egypt	Death	Virtue Faith Truth	Fertility Strength	Happiness Prosperity	Joy
India	Life Creativity		Prosperity Fertility	Success	Death Purity
Japan	Anger Danger	Villainy	Future Youth Energy	Grace Nobility	Death
China	Happiness	Heavens Clouds	Ming Dynasty Heavens Clouds	Birth Wealth Power	Death Purity

is reduced. Other ways that the collective trait is displayed is how family and friends can borrow the bankcard between each other. Some even use the ATM to transfer money between family or friends in distant geographical locations, having extra cards for the same bank account. Since the perception of a technology can differ considerably from what the designers are used to, finding out what the target users' attitude is can give additional insights to influence the design.

The issue of transferring a system developed in the U.S. to Europe was discussed at a company seminar [11]. The experience of the developers on the European side of the company was that they first have to spend time adapting the product to the local area. Before launch of the system, they must allow for different currencies, adapt for different laws, different ways to do things and other demands for reliability. The impression given was that this could have been better prepared from the original developers.

If designing e-learning programs, the culture of the learners must be taken into account. Because culture and learning are interwoven and inseparable tensions might arise if the teaching style of the software does not correspond to the learning style of the learners [12]. An example is if the learners prefer to learn by experimenting for themselves, by specific instructions from the teacher or by helping each other in a group. This would most likely also extend to software tutorials.

## 2.1 Examples showing the benefit of nationalization

As a motivation to spend money and effort on doing nationalization, a couple of examples are given below. First one where cultural differences was not taken into account, then one where they were.

LYRE is a French system which lets students view poems from different “viewpoints” constructed by the teacher. The students can then view annotations linked to different parts of the poem as well as adding more annotations. What was considered unacceptable in Denmark with the system was that the students could not add new viewpoints. Only the teacher could do that. While this limitation was considered unacceptable in Denmark, the possibility for students to add more viewpoints could have been unacceptable in France where the original design was not a problem. [13]

According to Russo and Boor [10], Lotus Corporation, a successful company on the international market, had close cooperation with Japanese consultants throughout the process of adapting 1-2-3 to the Japanese market. The developers thought it would be helpful to provide a way of changing the name of the Emperor since dates are kept in years of the Emperor. The Japanese consultants strongly advised not to do that: they should not even dream about marketing a program that appeared, in any respect whatsoever, to question the Emperor’s immortality.

### 3 How Nationalization Can Be Done

*Nationalization* is the process of making a computer application or website adapted to the local culture of its users. (*Localization* is another word sometimes used.) This involves translation of language, but also adaptation of layout, icons and other culture specific items. A more general process is customization, which also deals with concerns other than culture [4].

*Internationalization* is what is done to make nationalization possible. This can include extracting strings to a separate file that can be replaced with another for a different language. But there are other concerns as well, such as style and symbols, that need to be handled. If internationalization is done from the start, later nationalization and other customization will be easier and less expensive [4].

#### 3.1 Internationalization

Nationalization starts with internationalization. All culture sensitive objects, including text, symbols, layout etc., must be isolated so that they can later be replaced by appropriate substitutions [10]. In Java, this can be done by using the provided resource bundles [14]. A new resource bundle is made for each locale, if necessary built in a hierarchy of language and country specific variants of the language. An object is stored together with a key used for accessing the object. So a call for WELCOME-MSG may return “Welcome” in England and “Välkommen” in Sweden. Similarly, pictures, whole panels or other types of objects can be stored in the bundle.

Part of internationalization is to allow the translated text to expand. The expansion occurs at every level, from individual fields to windows. In general, short messages expand more (in %) than long messages when translated [15]. Messages should be written independently and not be composed of separate



parts to prevent ambiguous or syntactically erroneous messages [15]. Different currency, time formats and other country dependent units makes it necessary to allow the contents and format of fields to vary. Different reading directions influences where to put what sort of information. Giving the translator tools to see how the interface is affected by the translations can give the translator an opportunity to choose words that fit the overall structure and appearance better where possible [15].

What needs to be considered when internationalizing includes language, culture and standards, both national and international [4]. Not checking standards can be a severe problem when wanting to market a product elsewhere. For a while, when U.S. companies tried to market PCs in Germany it turned out to be impossible, because German standards required some ergonomic features in the keyboard that could not be fitted in the original design [4].

### 3.2 Anthropology

Anthropology can be used as a means for understanding the context in which an application is used. Räsänen and Nyce [16] argue that what is considered to be context should be more than the immediate workplace and organization as is often the case in HCI. The larger historical, socio-structural processes also affects how people behave. In the article is an example of employees in a Contact Centre. They are very observant on the number of incoming calls and the number of operators logged in to answer the calls. If they have some other task to do than answering the phone, they get disturbed by a heavy load of incoming calls and may cover the display. Still, they can't keep from checking the display from time to time. They feel they should help out lessen the burden on other operators and the waiting time for callers even though they have other tasks. The explanation given in the article was that the Contact Centre was outsourced into the archipelago. The employees had to fight to get the positions and thought they had to work hard to keep their jobs. Since they are also reframed away from power, there is a higher degree of individual responsibility to get the job done. So the history and culture outside the immediate work situation had a profound effect on the work. The more we know about the world the user lives in and acts on, the greater the chance that what we design will support the user in the work [16].

### 3.3 Cultural Attractors

A concept put forward by Smith, Dunckley, French, Minocha and Chang [17] is cultural attractors. Cultural attractors are the interface design elements that reflect the signs and their meanings to match the expectations of the local culture. Typical cultural attractors are colours, colour combinations, banner adverts, trust signs, use of metaphor, language cues, navigation controls and similar visual elements that together create a look and feel familiar to the users for that particular domain. Finding the cultural attractors can be done by picking successful local websites typical of the relevant or related domains. The sites are

then audited by a usability expert from or with a very good understanding of the target culture. Colours, signs etc. (as above) and the meanings they have in the local culture are documented. By doing this, designers can come to understand how websites are usually built in the region and how to convey a desired message, for example trust.

The aim for Smith et al. [17] is to build a collection of these cultural attractors for different cultures. These could then be reused in future localization projects, both for auditing other websites or software products and as parts of a specific design solution [17].

### 3.4 Cultural Dimensions

Hofstede [18] studied culture in the context of organizations and describes cultures in terms of cultural dimensions. *Power distance* is the extent to which unequal power is expected and accepted. *Collectivism-individualism* refers to if the individual or the group is more important and if one is expected to look after oneself or support each other in the group. *Femininity-masculinity* measures how strong the gender roles are. A feminine culture have less distinct gender roles. *Uncertainty avoidance* is only relevant for Western cultures [17] and is how threatening unknown situations or uncertainty are. For Eastern cultures [17, 19] there is *long-term orientation*, which is the extent to which long-term gain is preferred over short-term results.

Cultural characteristics can be used to guide design. Marcus and Gould [19] use Hofstede’s cultural dimensions to give implications for how to build websites for different types of cultures. An example based on the power distance dimension is to vary how highly structured access to information is depending on how strongly the people of the culture expect uneven power distribution. The different dimensions can vary greatly in significance [17], so all of the recommendations might not be useful for all cultures.

In a study about e-learning [12] the considerations are based on the same cultural dimensions. The degree of leadership from the instructors could be varied depending on the culture’s expectations on strong authority. Emphasis could be on collaborative work or individual assignments.

“Given the impact that culture has on people’s behavior, truly functional global e-learning systems should reflect the cultural orientation of its users and not just be a translation of an American interface. Conceptual localization that fits the user’s culturally specific mental model of the e-learning system with functionality, feedback, and support for learning is a much more effective way to design e-learning systems for global use.”  
[12]

### 3.5 Cultural Fingerprint

Smith et al. proposes the method of taking a cultural “fingerprint” to see how well a website match the local culture [17]. The value of four of Hofstede’s cultural

dimensions (power distance, uncertainty avoidance, individualism-collectivism, masculinity-femininity) is converted to a scale ranging from 0 to 10. For the country fingerprint, the values of the dimensions are based on Hofstede's scores or any other available better data. The website fingerprint is based on experts' evaluation against criteria based on Marcus' and Gould's [19] guidelines. Examples related to power distance can be the structure of information or prominence given to leaders. Up to five experts independently evaluate each site. Every dimension is given a value between 0-10, taken as the average between all the experts' opinions. While some dimensions may be important in a culture, others might not be significant at all. Because of this, the importance of each dimension is also visualized and will be the same for both culture and website. The fingerprints of the culture and the website are then compared to see how much the website correspond to the culture. It provides a simple means for communicating about the localization issues with designers, developers and users [17].

## 4 Conclusions

There are strong indications that nationalization should be done. It is not just a matter of whether the users can understand the English language or not. Other requirements such as level of control seem to be at least as important.

## 5 Discussion

The need for localization has been argued throughout the article, but what about if the globalization makes us used to how things are usually done, so that we can use computer applications well even if they are not adapted to our own culture [20]? To some extent this is probably true with standards evolving and the internet as a global medium to spread ideas. But as the examples above indicate, complete lack of localization is not likely to be an option in the foreseeable future. This is of course depending on the situation. Simple applications or websites targeted at similar cultures may not need to change much. The important thing is to make sure whether or not this is the case. But the differences between work cultures are greater than differences between national cultures, so it is still more important that the product is adapted to the task than to the culture [4].

Even though most web pages are mainly focused on textual information, many localization guidelines for website design and computer applications can probably be used interchangeably. Especially since more application-like content is moving out onto the web the difference is further reduced.

While cultural dimensions might be of good use for design implications, relying only on them can result in missing other important aspects, for example the Chinese bargaining habits or the Japanese Emperor. But if resources are scarce, it is a simple way to be at least a little bit better off. The problem is that there may be several cultures within a single country, so care must be taken when generalizing [4].

## References

1. Efendioglu, A.M., Yip, V.F.: Chinese culture and e-commerce: an exploratory study. *Interacting with Computers* **16** (2004) 45–62
2. Shneiderman, B.: Universal usability. *Communications of the ACM* **43**(5) (2000) 85–91
3. Preece, J., Rogers, Y., Sharp, H. In: *Interaction design: beyond human-computer interaction*. John Wiley & Sons, Inc (2002) 459 Interview with Ben Shneiderman.
4. Karat, J., Karat, C.M.: Perspectives on design and internationalization. *ACM SIGCHI Bulletin* **28**(1) (1996) 39–40
5. Peltokorpi, V.: Japanese organizational behaviour in nordic subsidiaries: A nordic expatriate perspective. *Employee relations* **28**(2) (2006) 103–118
6. Lindell, M., Arvonen, J.: The nordic management style in a european context. *International studies of management and organizations* **26**(3) (1997) 73–91
7. Jansson, H., Golubovic, O.: Globalisering eller lokalisering av elektroniska-handelsplatser över geografiska områden. Master's thesis, Göteborgs Universitet (2001)
8. Angeli, A.D., Athavankar, U., Joshi, A., Coventry, L., Johnson, G.I.: Introducing atms in india: a contextual inquiry. *Interacting with Computers* **16** (2004) 29–44
9. Nielsen, J.: (International usability testing) [http://www.useit.com/papers/international\\_usetest.html](http://www.useit.com/papers/international_usetest.html), accessed 2007-05-09.
10. Russo, P., Boor, S.: How fluent is your interface?: designing for international users. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. (1993) 342–347
11. Brothers, L.: (2007) Seminar with Lehman Brothers, 24th of march 2007, Umeå University.
12. Downey, S., Wentling, R.M., Wentling, T., Wadsworth, A.: The relationship between national culture and the usability of an e-learning system. *Human Resource Development International* **8**(1) (2005) 47–64
13. Nielsen, J.: Designing for international use. In: *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*. (1990) 291–294
14. O'Conner, J.: Java internationalization: Localization with resourcebundles (1998) <http://java.sun.com/developer/technicalArticles/Intl/ResourceBundles/index.html>.
15. Merrill, C.K., Shanoski, M.: Internationalizing online information. In: *Proceedings of the 10th annual international conference on Systems documentation*, ACM Special Interest Group for Design of Communication (1992) 19–25
16. Räsänen, M., Nyce, J.M.: A new role for anthropology?: rewriting "context" and "analysis" in hci research. In: *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*. Volume 189 of ACM International Conference Proceeding Series. (2006) 175–184
17. Smith, A., Dunckley, L., French, T., Minocha, S., Chang, Y.: A process model for developing usable cross-cultural websites. *Interacting with Computers* **16** (2004) 63–91
18. Hofstede, G.: *Cultures and organizations, software of the mind*. HarperCollinsPublishers (1994) Paperback edition. Original edition published 1991 by McGraw-Hill.
19. Marcus, A., Gould, E.W.: Crosscurrents: cultural dimensions and global web user-interface design. *Interactions* **7**(4) (2000) 32–46
20. Eriksson, M.: (2007) Personal communication. E-mail: eonmia03@student.umu.se.

# Social Interaction in Virtual Worlds

Göran Lundin

Department of Computing Science  
Umeå University, Sweden  
dit03gln@cs.umu.se

**Abstract.** Online computer games are a very large and still growing market. The world of online computer games allow for new channels of social interaction. This article describes how and in which forms social interaction takes place in the real world in face-to-face encounters. The article gives insight to earlier research conducted and finds that today's online computer games try to use the different kinds of social interaction that is available in the real world. In the last section, a short analysis of an online multi-player computer game is conducted to see what kinds of social interaction are possible in that game. The analysis shows how this game uses social interaction that is available in the real world and most commonly used by us humans in our daily lives. This is not a task without problems but it is solved to a, for the game, satisfactory level.

## 1 Introduction

Social interaction in the real world is complex, ranging from verbal language which we can control to facial expressions and body language of which some is almost out of our control. It is not an easy task to try to extend these ways of interaction and communication from the real world into the virtual worlds of Collaborative Virtual Environments and Network Virtual Environments. At the front in this work is the world of computer games and especially Massively Multi-player Online Role-playing Games. The developers have recognized the need for interaction among the players and that it is in letting the players interact that the success of a game lies.

This article investigates how it is possible to interact in virtual worlds today and which dimensions of interaction are afforded. Battlefield 2142 is one of the biggest Massively Multi-player Online Role-playing Games on the market today and a closer investigation on what different types of interaction and communication it affords is conducted.

## 2 Social Interaction

Social interaction is something that we do all the time, in almost all situations where there are other people present. "Interaction means actors take one another into account, communicate, and interpret one another as they go along." [1]. Talking to another person is interacting socially, giving someone a rose is

interacting socially, chasing someone is interacting socially. The list of ways to socially interact is infinite. In most social interactions we try to communicate a message, therefore it is possible to say that most of our social interactions are forms of communication.

## 2.1 Social Objects and Symbols

All around us in the world there are objects. In our everyday life we see them and do not think more about them. They are there as they always have been and always will be. Many objects have been created by nature and many have been created by man. But to a human an object is not just an object; it is interpreted and given meaning and a purpose through social interaction. As we learn what an object is and what its purpose is it becomes a social object. We constantly learn from each other what different objects are, which is why they are called social objects [1]. A social object is “any object in a situation that an actor uses in that situation. That use has arisen socially” [1]. But being a social object is not restricted to physical objects. What a social object is, depends on the situation we find ourselves in. If one person asks another person something, that person becomes a social object to the person asking the question. In similar ways animals and other living things can be social objects. Emotions can be social objects, we can define, interpret and use them in others as well as ourselves. We ourselves can also be social objects. Another class of social objects are symbols [1].

A symbol is a social object that represents something that, through social interaction, people have agreed upon that the symbol should represent. Thumbs up stands for good; flower power stands for rebellion [1]. Symbols are therefore “social objects used by the actor for representation and communication” [1]. It may seem that almost all social objects are symbols but that’s not the case. Take a flower for example. It may be for picking, smelling or eating. But not until it is given to someone to represent love does it become a symbol. Symbols are social because what they represent is agreed upon through social interaction. As with social objects there are many symbols. What we do and how we act can be a symbol, walking out of the classroom during a lecture can be a symbol representing that the lecture is bad. Many objects made by nature or man can be symbols, for example it is commonly agreed that a diamond stands for luxury. One of the most commonly used and important symbols is language [1]. Language is at the center for what a symbol is, because “it is a set of words used for communication and representation” [1]. A language is built up of words which are also symbols, they represent something else. Without words all other symbols, like acts and objects, would be without meaning to us. We would not be able to describe them without the help of words. So words are not just a symbol like other symbols they are the most important kind of symbols, without it we would not be able to agree upon the meaning of other symbols and what they represent [1]. Symbols are at the center of human communication and “is the basis for almost everything which catheterizes the human being in nature” [1].

## 2.2 Social Action

A social action is an action where the actor takes other actors into account and adjusts her actions according to them or their actions. Other actors make a difference in what we choose to do and how we choose to act. The actor uses other actors and who then become social objects to the actor. Almost everything we do is some kind of social action. To talk to or listen to, to hit or to kiss another person, all of these are social actions [1]. The most important aspect and the foundation for social actions is that “we take each other into account” [1].

Very often when we act and do things there is more to it than just doing it. More often we try to communicate something through our actions. Almost all social actions are symbolic to some extent. A kiss is not just a kiss, but a symbol that represents love. Hitting someone is not just a hit, but may represent dislike. In this way most of our social actions are symbolic, where almost everything that we do comes with an underlying meaning that we try to communicate through our social actions [1].

Social actions can be classified into different types. First there are *wert-rational actions*, these are actions that are taken because they lead to a valued goal. However no thought is given the consequences and the appropriateness of the means necessary to accomplish the goal. Second there are *zweck-rational actions*. Instrumental actions are carried out after thorough planning and evaluation and valuing goals against each other. Thorough consideration to the different means, and their consequences to achieve the goal are also taken. There are also *affectional actions*, which are social actions that are performed to express ones feelings. The fourth kind of social action is *traditional actions*. These are actions which are carried out due to tradition [2].

## 2.3 Communication

Social interaction in its most common and easily understood form can be described as communication. In our everyday lives we constantly use symbols to communicate with others. As have been presented there are many kinds of symbols. There are many channels this communication can be done through. For the purpose of this paper two interesting channels of communication will be brought to focus.

**Language** is our most important mean of communication. Without it not much would differ us from primates. We use it to communicate to others what we feel, how we feel, what we think and many other things in our daily lives. Our language is a kind of symbol where we construct sentences that consist of words that represent something. The actor sending the message puts a representation into to the message, but it is up to the receiver to interpret the message and understand what the sender wants to say. The symbolic representation of a sentence varies. In one situation the sentence ‘It is cold in here’ might mean ‘I want a sweater’, but in another situation it might mean, ‘Close the window’ [3].

It is common to differentiate between verbal and written language. When using verbal language it is possible to send more ‘rich information’, which means it is possible to put more information in other channels than just the spoken word [4]. One channel in which information can be added to a message when using verbal language is paralinguistics [3]. Paralinguistics are the sounds we make which are not verbal such as, pitch, rhythm, intensity and pauses. Paralinguistics do not deal with what we say but how we say it. A problem that can arise with verbal language in combination with other channels for communication is the possibility that the different channels of communication transmit different messages [4]. This makes it harder for the receiver to interpret the message and what the sender really means.

Written language is more limited in the possible information it can transmit. There are no forms of paralinguistics that is applicable to written language. The symbols and sentences transmitted as written language represent what they have been determined to represent. There is a clear limitation to the richness of the information transmitted through written language. Another problem with written language is that it usually takes more time to get a response to the information sent than with verbal language. Information that is more impersonal and meant for more receivers is usually sent using written language, while messages that are more private and to a smaller receiver group usually is transmitted through verbal language [4].

**Body Movements and Gestures** is another important channel to transmit information through. They can act as a complement to the verbal language. When involved in a discussion it is common for most individuals to gesticulate with their hands and body to transmit as much information as possible. This kind of nonverbal behavior that is tightly linked with verbal communication is called *illustrators*. It is also possible to use body movement and gestures as the only channel to communicate. These kinds of gestures are called *emblems*. The wave of a hand is widely recognized in our society as a greeting. Emblems are not without problems. A gesture that has one meaning in one society might mean something totally different in another; on the contrary one message might be transmitted by different gestures in different societies. This might lead to confusion among members of different societies when they try to communicate [3].

### 3 Virtual Worlds

Virtual worlds and environments are making good progress especially in the form of computer games. The Collaborative Virtual Environment (CVE) was presented in the late seventies in the form of Multi-User Dungeons or MUDs [5]. The MUD is a text based Collaborative Virtual Environment; it afforded people to meet and interact through a chat like environment. The advantage of MUDs are the almost infinite degrees of freedom for interaction among the participants [6].



The CVEs of today are mostly represented by graphical Virtual Environments (VEs). These provide another representation of a simulation of real worlds, places and actions [6]. A further expansion of the term CVE is the Networked Virtual Environment or Net-VE. A Net-VE is a world in which multiple users can interact although they may be located in different parts of the world [7]. There are five features that distinguish a Net-VE [6]:

1. A shared sense of space (illusion of being located in the same place)
2. A shared sense of presence (avatars of participants)
3. A shared sense of time (real-time interaction possible)
4. A way to communicate (various interaction methods)
5. A way to share (dynamic environment that can be interacted with)

In CVEs and Net-VEs available today there are great variations in technical features although there are some consistencies. There are five primary mechanisms for users to use when presenting themselves in the virtual environment. The actions available include, choosing a screen-name, the choice of an avatar, motion, user-constructed artifacts that can be interacted with and speech, text or audio based [6].

Computer games constitute a big and growing part of the virtual environments available today. One genre featuring virtual worlds is the Massively Multi-player Online Role-playing Games or MMORPGs. These games afford many forms of social interaction between the players and they have recognized that the success of online computer games lies in letting the players interact [5].

### 3.1 Social Interaction in Virtual Worlds

According to a study conducted by Steding et al [8] the most important aspect of a multi-player game, to the players, is the possibility to interact. As aforementioned the game genre of MMORPGs has addressed this fact and has made it the most important aspect. Without interacting with other players it is impossible to be successful.

**Verbal Communication** Moore et al [5] studied the MMORPG Star Wars Galaxies (SWG). The game is an entire 3D-world of its own depicting the world constructed in the Star Wars movies. Within the game players take the form of a character of their choice. There are many different characters. The successes of the different characters depend on the others and interacting with them. In Star Wars Galaxies the social, verbal, interaction among the players are purely text-based. They can either communicate only with other players in their vicinity by typing text messages, or if they have joined a group with other players over unlimited distances. There is also possible to program the character to perform a sequence of actions and at the same time display a certain message. This becomes a problem since the communication among the players gets flooded with communication that can be resembled with e-mail spam. This makes it harder to distinguish the messages that the player want to receive [5].

The form of text-based verbal communication originated with the MUDs and has evolved to what it is in today's MMORPG. With the progress made in technology today verbal communication is no longer restricted to the text-based form. Many games feature different kinds of audio communication. The most common kind are preprogrammed messages which the player's avatar makes at certain events in the game [9]. The game studied by Manninen [9] is the first-person-shooter game Counter Strike (CS). CS is a team oriented quite realistic military action game, where the player can adopt one of several characters, like a sniper or medic. Typical preprogrammed audio message in CS is crying for a medic when injured, alerting other players of the presence of the enemy or calling for backup. In later years, with the development of voice-over-IP communications protocols, the audio communication have taken a new dimension where players can talk directly to each other [9].

**Non-Verbal Communication** There is a large array of forms of non-verbal communication in virtual worlds. As aforementioned Manninen [9] has studied different interaction forms in the game Counter Strike, in his article he distinguishes between several different interaction forms. The first and fastest to interpret is the *appearance of the player's avatar*. The appearance of the avatar communicates to the other players the role of the player, e.g. if he is a medic or a sniper, and which team he belongs to. The different roles and team belonging is seen by the clothes and the equipment that the avatar carries. The appearance of the avatar is a very important kind of interaction since it is the first that tells a player whether the avatar in front of him is friend or foe. One problem with this kind of communication is that it becomes less accurate with distance, the further away another avatar is the harder it gets to recognize belonging and role dependent of appearance [9].

*Facial Expressions* is a difficult channel to use and is exclusively used with preprogrammed expressions connected with certain animation sequences, like getting injured or killed. A problem with using facial expressions in games like Counter Strike is the high pace, there is just no time or level of detail to be able to observe and interpret the expressions [9]. On the other hand in games like Star Wars Galaxies studied by Moore et al. [5] where role playing and more precise interaction play a bigger part, interaction through facial expressions is well developed. In SWG there are a range of different facial expressions the player's avatar can make, triggered by the player. It is an essential part of the multitude of interactions possible within the game.

*Kinesics* is another important way of communicating. It includes all body movements except touch and is commonly referred to as body language [10]. In SWG body language play an essential part like facial expressions. The avatar can make a large number of different gestures triggered by the player, like bowing and cheering, to communicate [5]. In higher paced games like CS the use of body language is more automatized. As with the audio messages most of the gestures and the communication with body language available is connected with preprogrammed animations, like limping when injured. Because of the lack of

gestures available the players of CS have developed their own in game gestures like moving the avatar back and forth to tell another player to move in that direction [9]. Also, the posture of the avatar is an important way of communicating, it tells teammates and other players something about what is going on in the closest vicinity. Specific movements of the head is something that is not supported in CS, it is coupled with the overall body movement [9].

The use of *non-verbal audio* is, in games like CS, an important communication tool for acknowledging what is going on in the vicinity of the player. Non-verbal audio include all sound effects like shooting, walking, reloading, etc. These effects can tell the player much about what is happening around him, if there are shots fired that might indicate that it is time to take cover and fire back. Silence is also an important factor to consider. When sneaking around the disturbance of silence, by walking on something that makes a sound, might be fatal [9].

*Physical contact* is an important interaction form in games like CS, but might not play such an important role in games like SWG since there is time for more precise forms of interaction, like written or spoken language. In CS the goal is to eliminate the opposition in any way possible. This means that close combat and physical contact, like stabbing or throwing grenades or shooting at each other, is common and unavoidable [9].

Overall it is possible to say that the non-verbal communication consists of many different dimensions which all contribute to shaping the experience of the game.

### 3.2 Problems with Social Interaction in Virtual Worlds

When using text-based verbal communication some problems might arise. As aforementioned by Moore et al. [5] there is a risk that if there are too much messages transmitted at the same time it becomes too much for the player to handle. Hence important messages will be missed. In games like Counter Strike text communication is just not fast enough and it steals too much attention. Manninen found in his research that usage of other text messages than pre-programmed where almost exclusively restricted to between game interactions. While playing the players were to occupied with staying alive they did not have time to write messages [9].

One thing that would heighten the interaction possibilities and make for richer interactive language in virtual worlds would be if it was possible to explicitly control different facial expressions and body movements. The biggest problem with non-verbal communication like facial expressions and kinesics are the lack of control possibilities. An ordinary keyboard does not contain sufficient controls to efficiently support the control of specific facial expressions and body language. It would take too much focus from the rest of the communication for the player that the risk is the player would lose the communication coming in while trying to control the avatar [6]. Also in games like CS which have a high pace there would not be time to control the avatar to the extent that is necessary.

## 4 Social Interaction in Battlefield 2142

Battlefield 2142 (BF) is a first-person-shooter game set 200 years into the future. Players can choose to play as a soldier for one of two military superpowers - the European Union or the Pan Asian Coalition. The game is a MMORPG and the goal for both teams is to conquer the lands of the other and win superiority of the field of battle.

When starting the game players are enlisted to one of the two armies. They then get to choose among four different roles, assault, reconnaissance, support or engineer, which one they want to play as. Each role has different equipment and weaponry suited for solving the task that the role is suited for. For each army there also is a commander which has the overall control of the army's joint resources, like artillery and unmanned-aerial-vehicle reconnaissance. Adding to the role of the commander there are also possible for the players to start a group of up to six players and by that become group leader. In addition to the overall goal it is possible for each player to earn individual points. As the player gets more points he rises in rank and earns more special equipment. The chance of getting points is better if playing as commander, group leader or in a group.

In the game there also exist a wide variety of vehicles that the players can control. All vehicles take two or more soldiers.

### 4.1 Social Interaction

In the game several different ways to interact with other players exist. Ranging from pure text-based chat communication to body language and verbal audio communication, in the sections below the different kinds of interaction within the game will be examined.

**Verbal Communication** is in Battlefield 2142, as in most other MMORPGs, divided in to two kinds of communication, text-based and audio-based. The messages can be divided into, automatically generated or player generated. In BF there is a well developed system for calling out automatically generated messages. A player can press a key on the keyboard and get an interface in which a wide selection of messages are available, ranging from saying thanks to crying out for a medic or alerting teammates of the approach of the enemy. When choosing an automatically generated messages it results in that the player calls out to the other players in the audio channel the message selected, as well as a corresponding text message, which appears on the screen of the rest of his teammates, see figure 1.

The other type of verbal communication is messages that are generated by the player. These messages can be divided into, text and audio messages. There are three different channels that a player generated text message can be communicated in, in BF. Either a message can be sent to all players, friend or foe, or to the same team, or if the player is part of a group a message can be sent to the members of the group. To generate a message the player press the designated



Fig. 1. The message field on the screen

key of the type of message he want to send and marker appears on the screen at which the player can write anything he likes. The process is exactly the same for all types of messages. The message appears on the screen, see figure 1, of the players it was meant for. In addition to text messages there is also possible for real time audio communication, over voice-over-IP. This feature is only available between players that are members of a group. If member of a group then a player can talk into a microphone and the message will be heard in the speakers of the other players in the group.

**Non-Verbal Communication** Battlefield 2142 is like CS, the game examined by Manninen [9], a high paced game, therefore most of the interactions that occurs are brief and in many ways automated. When interacting in BF much of it is done without using verbal language. As aforementioned symbols are all objects around us that we through social interaction have agreed upon what they should represent, and that we use to communicate [1]. In a game like BF there are many symbols, used to a wide extent.

There are many social objects in BF that are used as symbols. Since BF is a game where the sole purpose is to gain superiority over the battlefield many of the symbols communicate messages of bad things. Since BF tries to resemble the real world, the meaning of a symbol is not agreed upon in the game, it already exist from the real world. A bullet flying through the air is a symbol representing that the shooter want the target to die, the act of stabbing communicate the same thing. But not all symbols communicate bad things. BF is a team game and therefore interaction within the team and trying to help teammates is essential. There are many symbols and acts that communicate good things to teammates, e.g. the medic healing another player. The medic can also leave his medic kit in the terrain then it becomes a standalone symbol representing getting healed and receiving strength.

Most of the actions performed in Battlefield 2142 are social actions. As the game is a multi-player game it is impossible to play it alone and not take other actors into account. Most of the actions performed are done to try to outsmart the enemy and often in cooperation with teammates. Many of the social actions, like the symbols, communicate a message, often of wanting the enemy to die or if it involves a friend to heal him or act as a team with him in trying to defeat the enemy. Within BF not many actions carried out can be classified as wertrational actions, because performing such an action probably will lead to your death. The most common kind of action in BF is the zweckrational action. Many actions, like ambushing an enemy, are performed after an evaluation of the means to achieve the goal and their consequences. As for affectional actions they like wertrational actions might occur sometimes but should be considered rare since there is no time to express feelings. Since BF is an ever changing arena it is hard to differentiate any kind of traditional action, there is just not two situations that are the same.

Like in CS it is possible for the player to take different roles when playing. Roles are represented by avatars with unique equipment. This is the first way that players communicate among each other when trying to establish which part of the team they all play. It also communicates the most important message of the game, if the avatar seen is a friend or foe.

The use of facial expressions in BF is limited to, as in CS, preprogrammed expressions made by the avatar in certain situations, like when being injured. Other than that there is no time to use facial expressions as a means of communication since the pace is too high.

Kinesics, body language and gestures, is another mean of communicating which is implemented in Battlefield 2142. As with facial expressions gestures are only used in specific animation situations like when the player is injured and cries out for a medic, he waves his arm to let the surroundings know that it was he who called. As for body language it is more widely and actively used by the players. The most common and obvious use of body language is changing stance. The player can alternate between standing, kneeling or lying down. The different stances often communicate what is going on in the surroundings and what level of alert to be in. If a player is lying down it is probably because the

enemy is shooting at him, and then it probably is a good idea to hit the dirt. But if a player is standing up looking around it probably is a safer environment. As with other ways of communicating in BF the high pace does not afford any more complicated and extensive ways of communicating, like in Star Wars Galaxies.

As in CS the non-verbal audio is an important part of the communication in the surroundings. Sounds like gunfire or grenades exploding in the vicinity are a sure tell that there is a battle going on nearby. Hearing a bullet sweep by your ear communicates that someone is shooting at you. Since much of the verbal audio communication is automated the use of non-verbal audio becomes more important. One, a bit more different, way of non-verbal audio communication in BF is using your weapon to communicate with teammates. The most common scenario of this is when trying to get a ride and the driver does not hear the calls for a ride, shooting a few rounds at the vehicle probably will get his attention and make him wait for you. This might not be a way intended by developers but it is afforded in the game, since most vehicles is not damage by small arms fire, and commonly used.

The different kinds of non-verbal communication mentioned above probably does not cover all the ways that players communicate in BF, but they include the ways that the game support and that is intended by the developers.

## 5 Conclusions

Supporting in-game interaction is not an easy task. BF tries to encourage players to interact and work together as a team by letting them create groups in which there are more channels for interaction and communication. It is not an easy task to afford different and versatile ways of communication in a game like Battlefield 2142 where the pace is high. When there is no time for communication through verbal channels other ways of interaction takes a larger place. Much of the communication that takes place does so through automatically generated messages and through body language, most commonly through the stance and actions of a player.

Which type of interaction that takes place in a game depends on the goals. In games with a slower pace there is more time for more time-consuming forms of interaction like verbal communication through text messages generated by the players. In games with a higher pace, like BF, there just is no time for stopping and chatting a bit with a fellow player, which probably would get you killed. Instead interaction is limited to faster ways of communication, like auto generated messages.

One should not forget the type of interaction that takes place between foes. Not just the common ways of communication sends a message, being shot at sends just as a clear message as text. This kind of interaction is probably the most common kind of interaction in Battlefield 2142.

Battlefield 2142 affords many forms of interaction from verbally based audio communication down to physical contact like stabbing another player. All channels for interaction might not be used to the same extent by the players

and interaction is not performed in the same manner as in the real world. In the setting of the game the types of interaction supported must be considered sufficient to fulfil their purpose.

## References

1. Charon, J.M.: Symbolic Interactionism – An Introduction, An Interpretation, An Integration. Prentice Hall, Inc. (1995)
2. Wikipedia: Social actions (2007) [http://en.wikipedia.org/wiki/Social\\_action](http://en.wikipedia.org/wiki/Social_action), accessed 2007-04-13.
3. Deaux, K., Wrightsman, L.S.: Social Psychology. Brooks/Cole Publishing Company (1988)
4. Jacobsen, D.I., Thorsvik, J.: Hur moderna organisationer fungerar. Studentlitteratur (2002)
5. Ducheneaut, N., Moore, R.J.: The social side of gaming: A study of interaction patterns in a massively multiplayer online game. *CSCW* **6**(3) (2004) 360–369
6. Manninen, T.: Rich interaction model for game and virtual environment design. Oulu University press (2004)
7. Manninen, T.: Interaction in networked virtual environments as communicative action: Social theory and multi-player games. In: Sixth International Workshop on Groupware. (2000) 154–157
8. Steding, A., Öström, F.: Utveckling av framtida massive-multiplayer-online-roleplaying-games—från quests till social interaktion. Technical report, Department of Informatics (2001)
9. Manninen, T.: Interaction forms in multiplayer desktop virtual reality games. In: VRIC2002 Conference. (2002) 223–232
10. Manninen, T., Kujanpää, T.: Non-verbal communication forms in multiplayer game session. In: HCI 2002 Conference. (2002) 383–401



# The relevance of aesthetics to the success of the Apple iPod

Mathias Bergmark

Department of Computing Science  
Umeå University, Sweden  
masbek02@student.umu.se

**Abstract.** This article focuses on different aspects of aesthetics in Apple's iPod and mainly how this is perceived through the visual modality. On April nine 2007 Apple announced that the millionth iPod had been sold. Can the aesthetic experience of the iPod explain why this product has achieved such an enormous success? Firstly, the paper will explore and explain the area of aesthetics and how this is perceived by a human being. A theory which explains the cognitive response to an artifact is addressed which divides such responses into either an aesthetic or an analytical one. The usability of the iPod is also addressed and if this aspect can evoke an aesthetic experience among users of the widespread portable digital music player.

## 1 Introduction

This paper explores how a graphical user interface (GUI) can benefit from aesthetics considerations. Specifically it will discuss what relevance this aspect has for the success of Apple's iPod.

It is generally implied that aesthetics is a subjective field. That it is a topic with its roots in art and not applicable in the world of computers and science. But, I believe this has changed due to the evolution of software and new electronic devices. Even though the computer initially was created for arithmetic computation it has evolved into a medium. A medium in the sense that it is a platform for interacting with represented information. The work of Douglas Engelbart, who invented the mouse and word processing in the late 1960s, started this way of using a computer and Alan Kay with colleagues from Xerox PARC refined this vision during the 1970s [1]. The ideas of these pioneers were improved and productized by Apple with the introduction of its GUI. Apple introduced the "Human Interface Guidelines" (1987) which was an attempt to form a consistency in simple icons, menus and dialogues across all their different applications. The goal was to achieve a transparency in the GUI such that it without friction provided the user with information [1]. The iPod, made by Apple, is an immensely popular and widespread product. Apple integrates the use of a computer and the iPod through the iTunes software.

Aesthetics is something which humans come across in daily life whether they choose to think about it or not. Evidently, electronic products such as the iPod

are designed but do this mean that it must have aesthetics which can appeal users in to using it? This paper will explore different aspects of aesthetics in the iPod and investigate if this has to do with its success on the portable digital music player market place.

### 1.1 Vision and Perception

The way humans experience the world is truly a subjective and individual perception, even though humans possess more or less the same physiology [2]. To have knowledge of how humans perceive their environment, and more specifically an electronic product such as the iPod, is essential when analysing why and how humans experience product aesthetics. This paper mainly focuses on how humans visually can experience aesthetics in the iPod. This is due to the fact that an aesthetic response is most frequently stimulated by visual information [3].

Colin Ware [4] describes a simplified two-stage model for perception of visual information. Firstly, visual information are processed through large arrays of neurons. These are situated in the eye and in the primary visual cortex in the back of the brain. Neurons are individually selective and are specialised to detect different kinds of information, such as the orientation of edges or the color of light. The arrays of neurons are divided into subareas which work in parallel to extract these features [4]. The main criteria to enhance visual perception of information, at this stage, is to present it in a way that makes it easily detected by the largest neuron areas in the brain [4].

The second stage contains a sequential goal-directed process. When humans recognise objects factors such as visual attention and memory play a large part. Both the short and long term memory are used and a simplified explanation is that objects in our environment are processed and matched with properties stored in the memory. The main mechanism for what is perceived in a specific task depends on visual attention. The perception of objects is said to be done one at a time [4].

Humans have limitations in their perceptual resources. It is not possible for a human to perceive all of the information in its environment simultaneously. Because of this the mental processes filter information in unique ways. Research has shown that people can choose on what level in the perception process that this filtering of information shall occur. This mechanism is partly due to the fact that humans try to find the simplest way to solve a task. It is possible for a human to direct their attention when for example having a conversation in a noisy room. This filtering in the perception process is achieved by closing down different modalities [2]. Similarly humans can direct their visual attention at different objects, which partly can explain why some visual objects attract human attention more than others [4]. The answer to how and why humans direct their visual attention to some objects and not to other is complex. One explanation is based on previous memories and information stored subconsciously in the human brain that triggers emotions linked to this stored information. These emotions can for example explain why some objects or colours are more pleasurable to view than

others. A theory describes that there are hundred of mechanisms which determine an aesthetic response, and where some of these responses are driven from symbols which are stored in the memory [3]. Some of these responses are due to vestigial adaptations for detecting physical features that were useful in an evolutionary aspect, which humans today still can experience [3]. A more fully presentation of this theory can be read in section four.

## 1.2 Aesthetics

Aesthetics is truly an ambiguous word. This section will discuss the meaning of aesthetics and how this word can be defined in relation to product design. The meaning of aesthetics may differ from person to person and their explanations, if expressed, can be unclear. Often when humans perceive objects, like in the case of electronic products such as the iPod, words cannot really express the experience of them. Still, it is stated that aesthetics designates “sensuous knowledge” [5]. Aesthetics is the knowledge which humans obtains through his or her senses and not through knowledge processed and generated in the mind. This can explain why aesthetic experiences of designed products in general, or more specifically electronic products such as the iPod, are hard for users to verbally express. Why do humans like or dislike the design of electronic products? One must comprehend the meaning of a products gestalt to fully understand its aesthetics [5]. A products gestalt is described as the whole of the product that is more than the sum of its parts [5]. This means that for example choice of colour and form of a product is perceived as a whole and the totality could give the viewer the aesthetic experience. The aesthetics in design are the experiences which the products gestalt creates in the mind of a human [5]. Furthermore, signs are basically the core to how humans are able to communicate. Our environment is full of signs which we humans either try to find meaning to or not. Whether humans succeed or not we always try to find meaning in objects that are perceived. Evidently, manufactured products possess signs which humans try to understand and find its intention [5]. Semantics is the study of the message of signs [5]. The semantic function explains the way a product can communicate with a user. Through the products semantic function user understands how to use it. Basically the products signs, which build up the products semantic function, communicate the products functionality to the user [5].

Gestalt psychologists has since the early twentieth century spread the principles of visual organisation [6]. In the article “Another look at a model for evaluating interface aesthetics” the authors have used these theories as a foundation when attempting to produce a well-defined model for screen layout. These ideas can be applied on any visual medium. The model partly focuses on how to achieve balance, unity, proportion, and simplicity in a graphical interface. A screen with larger objects is for instance perceived as heavier than one with smaller ones [6]. To achieve balance in a design it is essential to provide an equal weight of screen elements left and right, top and bottom of the screen. It is also important to find an equilibrium in the screen design. This is done by placing the layout of objects in the center of the frame [6]. Unity is another aspect in

screen design which makes the objects on the screen seem to belong together. To achieve this the space between objects should be less than the margins surrounding the objects on the screen [6]. Section 5.1 describes how this is a feature of the design of the iPods Click Wheel. Proportion in design is something which is culturally dependent. Some cultures find beauty in proportions which others do not sympathise with [6]. Some usually pleasing proportions are for example: square (1:1), square root of two (1:1.414), golden rectangle (1:1.618), square root of three (1:1.732) and double square (1:2) [6]. Simplicity is another feature in screen design which can be achieved by optimising the number of objects on the screen while also minimising the number of alignment points [6]. Simplicity in design is an aspect which is addressed in different ways in this article relevant to the iPod.

## 2 The iPod

The iPod was not a result of a technological breakthrough, and in 2001 it was clear to Apple that they had something better than any other rival on the portable music player market [7]. Since the introduction, in November 2001, Apple has produced ten new iPod models. On April 9th 2007 Apple declared that the 100 millionth iPod had been sold. These ten released models includes five generations of iPod, two generations of iPod mini, two generations of iPod nano and two generations of iPod shuffle. The iTunes and the iTunes store has in its presence helped how tens of millions of music lovers manage, list and purchase their music. The iTunes Store features the world's largest catalog with over five million songs, 350 television shows and over 400 movies. Over 2.5 billion songs, 50 million TV shows and over 1.3 million movies has been sold through the iTunes store which makes it the world's most popular online music, TV and movie store [8]. Evidently, the iPod is not only a music player but also a product which have changed the music industry and the lives people live. Steve Jobs, Apple's CEO, said "iPod has help millions of people around the world rekindle their passion for music, and we are thrilled to be a part of that" [8].

Remarkably, Apple was not in the beginning a leading developer of the digital music technology. Actually, Apple was basically the last brand of computers to be equipped with CD burners and various portable digital music players were already on the market before the iPod even was an idea [7]. One reason for the breakthrough of the iPod could be the fact that it has been boosted by artists in the music industry. This has probably sent a signal to teenagers who have seen the iPod together with their idols. The artist Moby was from the beginning of the iPod history such a person. He said once "The kind of insidious revolutionary quality of the iPod is that it's so elegant and logical, it becomes part of your life so quickly that you can't remember what it was like beforehand." [7]. In short the iPod has become an icon which is partly a reason for its success. Even the design of the white headphones made an impression and became an secondary icon for the iPod [7]. Jonathan Ive, Apple's vice president of industrial design, said regarding the white headphones "I remember there was a discussion:

Headphones can't be white; headphones are black, or dark grey.". Apple thought that the uniform whiteness of the iPod was too important and chose this design [7]. The article "The Guts of a New Machine", published in The New York Times on the 30th November 2003, emphasizes that one must also be aware of that "the fanatical brand loyalty of its costumers is legendary"[7]. Information technology in general has evolved into environments which people live their lives through [9]. Humans do not only buy products based for what they do, but also for the value of what they represent in society. When purchasing a product people not only get a product but they also feel a sense of being a part of society [9]. The aesthetics in products today play a large part in convincing consumers to purchase them [9]. This aspect can probably partly explain why so many consumers have chosen the iPod.

The starting point of the iPod was not a constructed chip or a design. It all started with the question, "What is the user experience?" [7]. The rest of this paper will focus on the generation of the iPod and whether the users aesthetical experience of the product has contributed to its enormous success.

### 3 Perceiving Aesthetics in Design

An aesthetic experience is thought to be an immediate response [5, 3] and information is most efficiently perceived by the visual system. This is why most aesthetical responses are stimulated by visual information. But, evidently other modalities also have the possibility to generate responses to an aesthetic stimuli. For example, a familiar sound can elicit an aesthetic response [3]. The cognitive processes named are many times executed concurrently. Depending on what is perceived some cognitive processes take more or less time. For example, detection of light and motion are both immensely fast processes [3]. The most immediate aesthetical responses are the results of adaptations which can be seen as an effect of the evolution of the human being. These responses are called vestigial and can be applied to modern artifacts but are not useful in a reproductive manner [3]. Vestigial responses can probably explain why we find some artifacts of today aesthetically pleasing. The iPod has a glossy surface which, according to this theory, can be seen to take a few seconds longer to detect. The attraction to glossy surface has an evolutionary explanation [3]. The evolutionary perspective claims that the human brain is attracted to reflective materials as a result of the time when humans lived on the savanna. Water has always been a valuable substance when it alone only could reflect sunlight. This can explain why humans experience, due to this vestigial response, glossy surfaces as rewarding and therefore pleasing to perceive also in our days [3]. The same theory also states that there are hundreds of information processing mechanisms which give the resulting aesthetic response. Some processes are drawn from symbols which are stored in our memory. One example is that the immediate vestigial response of an artifacts physical attributes, like for example the glossy surface on the iPod, can be replaced by what the attributes symbolically mean to the user. Symbolic

information associated with artifacts can elicit an aesthetic response depending on a cultural- and experience based context [3].

The aesthetic response of the iPods glossy surface, sometimes called “the candy-colored look”, can as described above be evoked in many ways and on different levels in the cognitive process. The boundary between an aesthetic response and an analytical one is not a sharply one in this theory. It is although a difference in quality between a response given within a few seconds, and one that has been processed for minutes or even hours. The process that deals with retrieving symbols from the memory are generally more likely to require a longer response [3].

The immediate and involuntary aesthetic responses can also effect the further exploration of the artifact. Beauty is preferred over ugliness [3]. This also affects artifacts in the sense that if it invites to a positive aesthetic response this enhances its chances to be more fully investigated by a user [3]. Previously in this paper a model was addressed how to achieve an aesthetically pleasing screen layout [6]. When applying this theory on the hardware interface of the iPod some of these guides seem to be present in its design. A balance in the iPods design is created with the centred placing of the Click Wheel and the LCD display. The simplicity in the iPods design is created by the use of directness and singleness of form. A proportion found to be pleasing is the golden rectangle (1:1.618) [6] which the main proportion of the iPod (1:1.675) nearly possess. From a first glance humans are also unbelievably good at sensing strengths and rigidity of structures which is something that probably has evolved through human evolution [3]. In this aspect the iPod with its stainless steel back and stable form probably infuses a positive response of reliability and quality. John Maeda, author of the book “The laws of simplicity”, explains though that this choice of material, on the front and back of the iPod, also can evoke a sense of the product being delicate and fragile. The stainless steel back creates a sort of illusion that the thin glossy surface floats over the surroundings, he also notes that people seem to find cleanness and simplicity rather attractive [10]. Some of this can probably be explained based on the vestigial responses [3]. From a usability point of view John Maeda has summarized how to achieve simplicity into ten laws [10]:

1. REDUCE. This is the simplest way of achieving simplicity which is done by thoughtful reduction.
2. ORGANIZE. When organizing large systems it appears to be fewer.
3. TIME. If a user saves time during use of a system this is perceived as simplicity.
4. LEARN. Having knowledge makes everything much simpler.
5. DIFFERENCES. Simplicity and complexity are important and need each other.
6. CONTEXT. The things which appear to be important may not be nearly as important compared to everything around. It is important to know what is important for a user in a given situation.
7. EMOTION. To evoke more emotions are better than less.

8. TRUST. To achieve trust in a computer, where it knows a lot about you as a user, is something which achieves simplicity. This makes many daily task seem simple and the user can “lean back”.
9. FAILURE. Some thing can never be made simple where some things must still be complicated.
10. THE ONE. The main idea of simplicity is to subtract the obvious when adding the more meaningful.

Usability is addressed further in section 4.1. This is a aspect which is important and relevant to aesthetics perceived in a product such as the iPod.

If the iPod achieves simplicity in its usability- and hardware design a remarkable side effect is the enormous market of protective and decorative iPod accessories. One could ask if people are drawn to simplicity then why do they want to have these accessories? John Maeda proposes two reasons to this. The simplified aesthetics of the iPod evokes a concern for its survival and therefore people find it important to protect it. Also, people seem to have a reason for self-expression and can express it through these accessories [10]. Evidently, accessories give the consumer the benefit of expressing their individuality and their feelings for the iPod.

#### 4 Aesthetics of use

In the late 80's Apple introduced “Human Interface Guidelines”[1]. This work was an attempt to make GUIs more consequently structured to achieve a higher understanding for the user when interacting with the system. The book “Hertzian Tales: electronic products, aesthetic experience and critical design” declares the importance of aesthetics in electronic products. It explains that the largest challenge for designers of electronic products is neither the semiotic functionality nor the technical aspects but more to investigate how to express the aesthetics of the product [11]. This view and Apple's visions of design and electronic products may seem contradicting but there is an overlap. Within the usability of a product there is a aspect of “an aesthetic of use” which a user can experience through interaction with a product [11]. It seems that a use of both Apple's “Human Interface Guidelines” and the vision of “aesthetic of use” is necessary to achieve a successful product. This can be linked back to Apples iPod when trying to explain its widespread success. Many electronic products can be pleasing to view but when interacting with the product this does not match the users' intentions and the result is an irritating experience. Products should possess both “beautiful appearance” and “beautiful interaction” [12]. Also, the usability of a product not only should be thought of as ease of use. A user may choose to work with a product despite it being difficult to use because it is challenging, playful, surprising, memorable, seductive or rewarding [12]. In relevance to this resonance in products are important [13]. Affordance, that gave rise to resonance, is a theory of Donald A. Norman which describes how the perceived and actual properties of a product gives information of how it could be used [14]. Resonance

is an individual experience where temptation, intimacy and engagement during an interaction is essential to increase resonance [13]. In aspect of making products more challenging, playful, etcetera this is truly an individual question when it depends on the person, the situation and the task [13]. Some people evidently prefer products which provide them with the result which they expect after an interaction where others like more a challenge. Still, natural mapping between product appearance, interaction and resulting feedback is essential for achieving resonance in a product [13].

Evidently there is more to it than appearance when analysing how aesthetics is perceived by a user of a GUI or a electronic product as a whole. The iPod is an good example of a product which possesses many of these different aspects of aesthetics. The subsection bellow attempts to clarify how the iPod uses usability and simplicity in its design to achieve a beautiful appearance and interaction.

#### 4.1 Usability

When viewing the iPod a sense of simplicity becomes present. The front has a flat glossy surface which has been widely copied due to its popularity. The back is covered by stainless steel. This same sense of simplicity is found when the iPod is connected to a computer with the software iTunes running on it. The music flows seamlessly through the FireWire cable into the portable player. On the front of the iPod a Click Wheel is placed which the user can navigate through lists of songs, artist, genres or playlists etcetera displayed on a LCD screen above. With a touch of a button the user is able to select items in the lists and use the Click Wheel once again to adjust the volume. Navigation is executed through the two main actions click and scroll. The Click Wheel is used by sliding a finger around the round surface. The Click Wheel is also divided into four sections which offer click button functions such as “Menu”, “Forward/Fast-Forward”, “Play/Pause” and “Back/Rewind”. Each button is positioned at 90, 180, 270 and 360 degrees respectively on the Click Wheel surface. In the centre of the Click Wheel a confirmation button is placed.

Through the five generations of the iPod the design of the Click Wheel has evolved. The first Click Wheel was designed as seen to the left in figure 2. In later versions some new designs were made and Apple separated the four buttons surrounding the Click Wheel and placing them into a discrete row of buttons. This design, seen in the middle of figure 2, made the iPod express a more complicated look. This design was later renewed into today’s more single and seamless control, seen to the right in figure 2. Also, a sense of unity [6] of the Click Wheel is more present in this design. An explanation to the redesign can be found in gestalt psychology . Gestalt psychologists believe that the human brain consists of mechanisms which tend to find patterns in objects [10]. Humans tend to group things together and even filling in blanks when for example viewing a box that is not fully closed [5], please see fig 1.

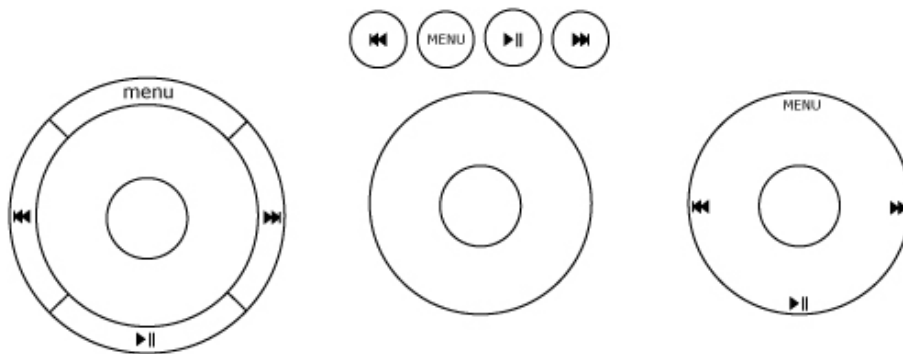
Humans tend also to organize and categorize what they see [10]. Johan Maeda explains “The principles of Gestalt to seek the most appropriate conceptual “fit” are important not only for survival, but lie at the very hart of the discipline of





**Fig. 1.** Gestalt law: Enclosedness. The viewer fills in the gap in the box [5].

design” [10]. When conducting a test John Maeda found a difference in how users interact with these different designs of the Click Wheel. In figure 3 one can see three diagrams which presents the differences in the interaction schemes for each Click Wheel respectively. The right diagram displays a cloud which shows how the individual elements of the Click Wheel are melted into one blurred cloud [10]. The right diagram illustrates from a gestalt law point of view a more simplified and integrated control. Evidently it can be a problem to interact with the Click Wheel when it is possible by mistake to hit the integrated buttons on the Click Wheel. The aesthetics of the blur found in the right diagram is common in the history of art [10]. Impressionist paintings by Monet with its hazy clouds of small brush strokes and more stylized images by artist Georgia O’Keeffe have an inviting nature similar to the one found in the right diagram of figure 3. John Maeda argues that the latest design of the iPods Click Wheel “blurs all controls into one image of simplicity”[10]. Here is an example of an “aesthetic of use”[11] where the use of gestalt laws in the design of the Click Wheel helped the iPod to achieve this aspect. The Click Wheel was from the beginning a new form of how to interact with a electronic product. The first two Click Wheel models do not express the aspect of an “aesthetic of use” in the same extent as the newest edition of the Click Wheel. This can probably not explain the initial success of the iPod.



**Fig. 2.** Illustration of the three designs of the Click Wheel [10].

Many companies have made jukebox software and others portable players. What Apple created with the iPod was both. Though this was not a question of doing more, but rather doing less [7]. Apple’s vice president Jonathan Ive



Fig. 3. Diagrams for the three Click Wheels in figure 2 [10].

states the iPod to be “overt simplicity” which is the most exciting thing about the product [7]. Jonathan Ive explained “What’s interesting is that out of that simplicity, and almost that unashamed sense of simplicity, and expressing it, came a very different product. But difference wasn’t the goal. It’s actually very easy to create a different thing. What was exciting is starting to realize that its difference was really a consequence of this quest to make it a very simple thing.” [7].

Usability is a tool to analyze if for example button labels make sense to a user. Information should be tested to see if it is grouped in the right category and if items are placed where user of the system might look when searching after it. Usability also tests the effectiveness of the completion of tasks [15]. The iPod’s clean gestalt and usability has made simplicity hip. Sometimes this can become a paradox when electronic products should have simplicity, but still offer the user features which involve more complex interactions [10]. Bruce Claxton, president of the Industrial Design Society of America (IDSA) 2003, gives a different and more or less contradicting view of usability “People are seeking out products that are not just simple to use but a joy to use.”[7]. As stated above this might imply that users want to interact with products which are a bit difficult to use because of the fact that it brings a sense of joy to it [12]. In aspect of design Steve Jobs, Apple’s C.E.O, emphasises usability is the main goal of the design when saying “Most people make the mistake of thinking design is what it looks like” and furthermore when stating “People think it’s this veneer, that the designers are handed this box and told, Make it look good! That’s not what we think design is. It’s not just what it looks like and feels like. Design is how it works.”[7]. To strive for a high usability factor is still the goal but maybe not the foremost one. Still, simplicity in iPods’ hardware, software and usability design were the main focus when designing it. The fact that the sense of simplicity is present in both the usability of the hardware- and GUI design probably enhances this experience of the iPod. The sum of these parts probably adds up to a gestalt of the iPod which a user perceives as aesthetically pleasing.

## 5 Discussion and Conclusions

An aesthetic experience can be elicited in many different ways and this is truly an individual perception which can be based on hundreds of mechanisms [3]. In this article it has argued that vestigial responses can partly explain why users of the iPod appreciate its glossy surface [3]. This is a key issue why aesthetics is important to consider when developing a product used by humans in general but more specifically the iPod. In relevance to this it was stated that the immediate and involuntary aesthetic response has also shown that if it is positive this can result in further exploration of an artifact [3]. In my opinion this can definitely have resulted in why so many people have chosen to purchase the iPod. A theory of a cognitive response of an artifact was presented which shows the complexity of the human perception. We have seen that an aesthetic response is most frequently stimulated by visual information basically because this modality provides data both more immediately and at higher rates than any other sense [3]. This is why this modality was the main focus of this paper when analysing how aesthetics in the iPod is perceived by a user. The iPod is a physical artefact and it would have been interesting to reflect over how the haptic modality perceives aesthetics in the iPod. Especially the Click Wheel would have been an interesting in this aspect. As stated there are hundreds of mechanisms which can give the resulting aesthetic response of an artifact. Even a cultural- and experience based factor adds to the equation which colours the resulting aesthetic response of an artifact [3].

Aesthetics can partly explain why the iPod has become such an immensely success but there is more to it. In this article we have seen that people not only buy products for what they do or aesthetically express, but also for what they represent in society [9]. The way Apple promoted the iPod when incorporating artist from the music industry resulted, in my opinion, to a large part of its success. This aspect is important but exceeds the scope of this paper. Still, this issue must be taken into consideration when analysing the success of the iPod.

Moreover the usability of the iPod takes great part in the aesthetical experience of the product. As stated, the goal of the iPods design was based on the user experience [7], where usability is an important part of the product. The achieved usability is due to the simplicity of the latest hardware design of iPods Click Wheel. It was noticed that people seem to find cleanness and simplicity rather attractive [10] which can prove why the iPod is perceived as aesthetically pleasing. The way that the iPod has incorporated this simplicity in its hardware, software and usability design has resulted in a unified product. The sum of these different design parts adds up to a gestalt of the iPod which the user may perceive as aesthetically pleasing.

## 6 Acknowledgments

I want to thank Håkan Gulliksson for all constructive feedback and help throughout the work of this paper.

## References

1. Jay David Bolter, D.G.: Transparency and reflectivity: Digital art and the aesthetics of interface design. P. Fishwick (ed) *Aesthetic computing*, MIT Press (2004) 1–7
2. Arbetskyddsmyndigheten: *Arbete-människa-teknik*. (1997) 175–195
3. Ulrich, K.T.: Aesthetics in design. In: *Design: Creation of Artifacts in Society*. (2006) Chapter 5
4. Ware, C.: *Information Visualization: Perception for design*. Academic press, San Diego, USA (2000)
5. Monö, R.: *Design for Product Understanding*. Liber, Liber AB, S-11398 Stockholm (1997)
6. David Chek Ling, J.G.B.: Another look at a model for evaluating interface aesthetics. *Int. J. Appl. Math. Comput. Sci.* **11**(2) (2001) 515–535
7. Walker, R.: The guts of a new machine, published: November 30. *The New York Times* (2003)
8. Press contacts: Apple: Tom Neumayr email: [tneumary@apple.com](mailto:tneumary@apple.com), A.C.M.e.: 100 million ipods sold. <http://www.apple.com/pr/library/2007/04/09ipod.html>, Press Release: Cupertino, California- april 9, 2007 (Published: November 30, 2007)
9. G. Crampton Smith, P. Tabor, T.W.: The role of the artist-designer. In: *Bringing Design to Software*. (1996) 37–58
10. Maeda, J.: *The Laws of Simplicity: Design, Technology, Business, Life*. The MIT Press, 55 Hayward Street, Cambridge, MA 02142 (2006)
11. Dunne, A.: *Hertzian Tales: electronic products, aesthetic experience and critical design*. MIT Press (2000)
12. Overbeeke, K., Djajadiningrat, T., Hummels, C., Wensveen, S.: Beauty in usability: Forget about ease of use! In: W.Greens and P.Jordans (Eds), *Pleasure with Products, beyond usability*. London: Taylor and Francis (2000) 1–10
13. Caroline Hummels, Philip Ross, K.O.: Human-computer interaction:interact'03. In: *In Search of Resonant Human Computer Interaction: Building and Testing Aesthetic Installations*. (2003) pp.399–406
14. Norman, D.A.: *The Design of Everyday Things*. The MIT Press, London, England (2001)
15. Alan Cooper, R.R.: Goal-directed design. In: *About Face 2.0 The essentials of interaction design*. (2003) 5–20

# Teaching Project Management Using Computer-Based Learning Tools

Jonas Bergström

Department of Computing Science  
Umeå University, Sweden  
dit03jbm@cs.umu.se

**Abstract.** This paper examines how computer based learning tools can be applied to project management. Project management is identified as consisting of three key concepts: Project Model knowledge, Planning, and Motivating the project group. The field of computer based learning tools, called E-learning, is examined. Simulations play a big role in E-learning and four different kinds of simulations are described: Branching Stories, Virtual Spreadsheets, Game-Based Models, and Virtual Products. Computer Games with pedagogical content, called Serious Games, is identified as a mix of some or all of the different simulation types. Three different tools for simulating project management are examined. It is found that none of these models offer training in motivational techniques. In the final section, e-learning techniques are applied to a sample project management curriculum. Web distributed learning objects are suggested for teaching project models, Serious Games are suggested for teaching work planning, and Branching Stories are suggested for teaching motivational techniques. The Branching Stories can be part of a Serious Game or distributed separately on a website.

## 1 Introduction

The computer can be used to support learning in a wide variety of ways. From flight simulators to army training, the list of applications is long. This paper examines the field of computer based learning tools and how such tools can be applied to the subject of project management.

The first section examines a sample curriculum that describes how project management is taught in a traditional fashion. In the second section, learning with the help of computers is examined. The strengths and weaknesses of various computer based learning techniques are discussed. The third section covers some existing computer based learning tools for project management. The fourth and final section shows how computer based learning techniques can be applied to the sample curriculum described in the first section.

## 2 Teaching Project Management

This section describes a sample curriculum for teaching project management in a traditional fashion [1, 2]. Key concepts of project management are identified as well as some problems with the traditional teaching methods.

The course is divided into two main parts. The first part consists of classroom lectures that teaches project models and practices often used in projects [1]. The second part consists of a sample project where the students get to plan a fictive project in a small group. The course gives training in project models and in work planning, but a project manager also needs to motivate the project group [3]. The course material presents some guidelines on how to do this, but it is not practised within the curriculum [1, 2]. Many of the concepts related to project management are hard to understand if they are not practised or experienced [4]. Digital learning techniques, and simulations in particular, can be used for providing experiential learning. This is the reason why digital learning techniques can be of assistance in teaching project management.

## 3 E-Learning

This section presents the field of computer mediated learning in general. Later sections will show how these concepts can be applied to project management in specific. Computers have a lot to offer when it comes to teaching and pedagogy. A computer game can make a learning experience more fun, a website can be used for distributing content, and advanced simulations can be used for training. The term E-learning is a collective term used to describe all these different forms of computer aided learning tools. The term learning object is often used in conjunction with E-learning but its definition seems to be as broad as E-learning itself [5]. A learning object can be described as content that facilitates learning and is accessed through a computer. For example, a learning object could be the training received from talking to a fictive character in a computer game or it could be a lecture distributed on a website. Note that the term e-learning does not necessarily imply interactivity. Making content available on a website can be considered as E-learning, even though it is not an interactive channel.

It is hard to find studies that show that students who learn through digital media achieve better results than students who learn through conventional classroom teachings [4]. Nonetheless a lot of money is being spent on e-learning. This confidence in e-learning comes in part from the various benefits that e-learning can provide aside from increase performance in learners, mainly its cost effectiveness and its ability to easily distribute learning content across geographical boundaries. Strother describes many success stories where companies have saved a substantial amount of money on their educational programs due to e-learning programs [6].

### 3.1 Simulations

Learning through computers is often in the form of simulations. However, a simulation does not have to be an advanced computer game or flight simulator. It can be a very simple model of a task, event or object. Aldrich divides simulations into four different categories which are described in the following sections [7]. All these different simulation categories can be combined to construct more advanced simulations, which are discussed in section 3.2.

**Branching Stories** A branching story is a series of multiple choice questions that simulates a complex task. A branching story could simulate a conversation with a person or simulate the assembly of a complex structure. Each choice made along the way influences the possible future choices until the end of the simulation is reached. The end result can be successful or unsuccessful to various degrees depending on the choices made by the student. The strength of the branching story lies in its simplicity. It is easily deployed and filled with content.

**Interactive Spreadsheets** In this simulation type the student tries to balance and configure various parameters to achieve a desired outcome. For example, an interactive spreadsheet could simulate a business economy with parameters including money spent on advertising, stock, and prices. This type of simulation is good for modelling complex tasks and is often carried out in teams with or without facilitators.

**Game-Based Models** In a game based model, the learning content is dressed up in some familiar game context. It could be a trivia game where the learning content is expressed as answers to trivia questions. Aldrich describes game-based models as more diagnostic than instructive [7]. A game-based model is not to be confused with a pedagogical computer game, which is described in a later section.

**Virtual Products** A virtual product is a slightly more advanced form of simulation. In this type, the student is presented with a simulated object designed to convey the look and feel of a real object. It could be used to familiarize the student with the inside of a car, or to get the student to understand a complex interface.

### 3.2 Serious Games

Serious games is a term used to describe Computer Games with pedagogical content and with an intent to teach and entertain. This can be a combination of all four of Aldrich's simulation types. Why would you want to package learning content into a game? According to a study by Webster, labelling of work tasks as play can increase performance [8]. By packaging learning content into a computer

game, which is associated with fun, and thereby labelling the content as play, it is hoped that the student will be more receptive to the learning content and perform better. Stapleton describes learning through computer games as “learner centred” whereas traditional classroom teaching is “teacher centred”. In a computer game, the learner, not the teacher decides what and when she or he learns [9]. Computer games are by definition highly interactive, and interactivity can also be claimed to support the learning process, as shown in a study by Khalifa and Lam [10].

Not all game types are equally good for serious games. In a study by Amory, it is shown that adventure games and strategy games are best suited for serious games [11].

**Adventure Game** The focus of an adventure game lies in exploring a storyline, solving puzzles and interacting with fictive game characters. The player takes the role of one or more player characters and guides these characters through the storyline. In contrast to the shoot-em-up, the adventure game usually does not require precise motor skills and reflexes. The adventure game often has a lot of dialogue and textual content that provides the player with clues, sets the mood for the game, or advances the storyline. Many adventure games also allow the player to develop the player character over time, gaining new abilities and opening up new possibilities in the game. The adventure game can be said to be an advanced version of Aldrich’s branching story.

**Strategy Game** In a strategy game, the player usually does not control a specific character. Instead, he or she acts as a supervisor or controller of resources. The goal of the game is to apply resources in the right amount at the right places and to make vital decisions that affect the flow of the game. Some strategy games do not have obvious endings (the popular *Sim City* for example), and some strategy games are played out against opponents and end when the opponent is defeated (*Starcraft* for example).

## 4 Existing E-learning Tools for Project Management

The following sections describe some existing e-learning tools for teaching project management. All of these tools come from the field of software project management, and are examples of Serious Games described in the previous section.

### 4.1 SESAM

SESAM stands for Software Engineering Simulation by Animated Models and it is a text based computer game designed to simulate software project management [12]. The game resembles an adventure game as the player takes the part of software project manager and interacts with a team of fictive developers in order to finish a project on time. The simulation is action driven, meaning that the player does something and receives the result of the action back. The game



permits several different actions to ensure that the project is concluded on time and within project specifications. When the game is finished, the player receives a score indicating how well she or he performed in managing the project. When finishing the game, the player gets access to a number of charts indicating how the choices influenced the outcome of the project.

This simulation builds on a dynamic simulation engine designed to be able to model various different projects. The model consists of two key parts: a static part and rules. The static part of the model describes entities in the simulation and the relationships between them. The rules part describes how these entities can interact and what the outcome of the interaction is. The state of the simulation is represented by the game state. As the player makes decisions and supply the model with input the entities affect each other according to the rules and the game state changes. Some, but not all of the information describing the current game state is provided to the player in natural language in the textual interface. The game state is transcribed to natural language using a dictionary which can be substituted and allow for simulations in different languages.

The data used in the simulation parameters is gathered from over thousand of real life projects [12]. To make sure that the result produced by the simulation corresponds well to reality, the SESAM model was compared with evaluation techniques used to estimate costs in a project. It was found that the data produced by SESAM corresponds well to the data produced by the evaluation techniques, and thus that the model produces realistic data [12].

Tests conducted by the constructors showed that SESAM could not be said to produce a measurable increase in performance in learners [12].

## 4.2 The Incredible Manager

The Incredible Manager is a computer game designed to simulate software project management [4]. The player of the game plays the part of a software project manager and has to manage a software project through five different phases: Begin Phase, Project Planning, Planning Acceptance, Project Execution and End Phase. In each phase the player makes vital decisions that affect the future of the game. For example, in the Project Execution phase the player can hire or fire developers, increase or decrease their workload and perform various other controlling tasks. The player has limited time to complete each phase. The player interacts with the game through a graphical interface, and some of the feedback on decisions is conveyed to the player through visual effects. The layout of the game resembles a strategy game.

As with SESAM, the simulations presented build on a dynamic simulation engine. The simulation engine of The Incredible Manager is built on the System Dynamics modelling discipline [4]. The System Dynamics modelling discipline will not be discussed in the present paper.

Evaluation of the game showed that students viewed the high difficulty of the game as motivating [4]. The visual feedback and the time constraints were also seen as positive. The downside of the simulation was its inability to represent some situations realistically. Students also reported that they would have liked

more feedback to show them how specific choices made during the simulation affected the outcome.

### 4.3 OSS

OSS stands for Open Software Solutions and it is a computer game that resembles an adventure game [13]. The player plays an employee at the fictive software development company OSS and can choose to participate in four different projects that the company is involved in. Each project presents a different task that aims to teach a certain concept common in software projects. For example, one task concerns certification, another concerns preparing state charts for components.

The game environment is graphical and highly interactive. The player guides his character through different floors of the software company and can talk to and interact with the characters that she or he encounters. Objects in the game world can be interacted with as expected by its real world counterparts.

This simulation focuses more on the work that is done within a project rather than the work done by the project manager. In each task, there is a game character that acts as the project manager and provides the player with the tasks that are to be done. Since the case studies used in this simulation are pre constructed it does not allow instructors to construct their own scenarios and projects. It is less dynamic than the simulations discussed above. However, the game presents a more realistic environment to the player, something that was found lacking in other simulations. Informal evaluation of the simulation revealed that some students thought that playing the game was too time consuming [13].

## 5 Problems

This section summarizes the problems and limitations with the E-learning tools described above and presents other E-learning issues to be considered when applying E-learning to project management.

OSS takes too much time and is not dynamic [13]. The Incredible Manager does not model reality in a satisfying way, and does not provide enough feedback to the player [4]. SESAM lacks graphical output and it is hard to configure the simulation model [14]. None of these simulations offer training for motivating the workforce, which is considered a key part of project management [3].

According to Aldrich, simulations as an educational tool are only effective with 80% of the learners. The rest will still perceive the task as an ordinary assignment and will try to please the teacher instead of embracing the simulated situation. When the simulated situation is not accepted, the learning experience will not be as good [7]. Simulations are not good as a stand alone concept, but should be applied as a complement to ordinary classroom based teachings [7, 12].

## 6 Enhancing a Sample Curriculum with E-Learning

The following sections describe how E-learning techniques can be applied to the sample curriculum described in section 2, to enhance the learning experience. Project management was identified as consisting of three key concepts: project model knowledge, planning activities, and motivating the project group. The following sections give examples of how e-learning can be used to support each of these key concepts.

### 6.1 Project Models

The first concept, teaching and learning project models, can be supported by learning objects distributed over the web. The content is stored in a database and accessed by students either directly in the browser as text-documents or as downloadable files to be stored on a personal computer and viewed at any time. The benefits of providing learning objects using a website is the accessibility and flexibility that it provides to the students [6]. It is also a cost effective alternative for the institution providing the course, since it does not require as much teaching resources in the form of instructors and classrooms [6]. There is no clear evidence that e-learning in this sense affects the performance of students, but since it does not seem to decrease performance, the benefits make it a very strong complement to traditional classroom teachings [6].

### 6.2 Planning and Scheduling Work

The second concept, planning and scheduling work, can be assisted by e-learning in the form of a Serious Game. The computer game is designed as a strategy game and is based on Aldrich's interactive spreadsheets. The computer game can and should draw from many of the concepts that exist in the Serious Games described in section 4. The student plays the role of project manager and is responsible for a fictive project and needs to allocate resources and plan work to meet various deadlines. The game can be turn-based, with each new turn representing a change in events or the completion of a planned activity. Alternatively, the game could be played in semi-real time with minutes representing days in the project. Semi real time can make the game more interesting [4]. The computer game can be played directly in the browser, or be available for download.

A computer game is a good way of teaching a student how to plan and schedule work, since it gives them the opportunity to try different courses of action and see the end result for themselves, instead of just reading about them. Drappa and Ludewig say that the best motivation for learning project management comes from seeing how bad management leads to the failure of a project [12]. In a computer game, the student gets to experience both success and failure in a captivating way, and without the immense costs that failure can bring in real life projects.

An important aspect when designing this computer game is the adequate modelling of reality. This can be done in two different ways. The computer

game could be designed in a static fashion, with the tasks, actions and outcomes modelled on real life examples, as it is done in OSS [13]. This means that the instructor will not be able to change the simulation to suit hers or his personal needs, but the simulation will provide an accurate model of how a real life project would run. The other approach is to design a dynamic simulation engine which allows the instructor to program the computer game to simulate different kinds of projects. This requires a very sophisticated simulation engine. The SESAM simulation engine and The Incredible Manager simulation engine both have there advantages. SESAM is more powerful and can model more situations that would occur in a real life project, but it is hard to program. [14]. The Incredible Manager aims to provide an easier way of programming the simulation, but has received critique that it can not model some situations realistically [4].

### 6.3 Motivating the Project Group

The third concept, motivating the project group is perhaps the hardest concept to teach. Here e-learning can be of assistance in offering dialogue training in the form of Aldrich's branching stories. The branching story simulates a dialogue with a member of the project group and the student gets to choose from different alternatives to try and motivate the fictive character.

The branching stories can be made available on the web on the same website as the previously mentioned learning objects. Several branching stories can be used to simulate different kinds of motivational dilemmas.

The branching stories can be tied with the computer game to provide a more complex simulation. This turns the Serious Game into a hybrid between strategy game and adventure game, since the focus now lies on resource management and interaction with fictive characters. If the player performs well in motivating the fictive project group, the group will work better, and deadlines and important milestones in the project will be easier to reach in time. The allocation of resources will become easier.

Branching stories is a good way of modelling dialogue [7]and motivating the workforce is naturally done by talking to the personnel.

All learning objects mentioned above should also be made available on cd-rom or some other kind of portable device to enable users who lack internet connection to take part of the learning objects.

## 7 Conclusion

There is little evidence that e-learning is superior to traditional classroom teachings when it comes to increasing performance in learners. But e-learning has other relevant benefits, mainly providing an efficient way of distributing content to learners. When teaching project management, which is a difficult task with many parts that need to be practised, e-learning can be used as an excellent complement to traditional teaching methods [13, 4, 12]. There are many existing tools for simulating project management today, and all of them have their

strength and weaknesses. The simulations presented in the present paper give no training in motivational techniques. Training in this area can be supported by branching stories, which can be integrated in a Serious Game or distributed as stand alone learning objects through a website. When designing a simulation for a project management process, one of the big challenges is designing simulation rules that are dynamic and flexible but at the same time model the project realistically.

## References

1. Umeå Universitet: Projektledning, curriculum (2007) [www.umu.se](http://www.umu.se), accessed 2007-04-12.
2. Tonquist, B.: Projektledning. Bonnier utbildning AB, Sveavägen 56, Box 3159, 103 63 Stockholm (2005)
3. Lewis, J.P.: Fundamentals of project management : developing core competencies to help outperform the competition. AMACOM, New York (2001)
4. Dantas, A.R., de Oliveira Barros, M., Werner, C.M.L.: A simulation-based game for project management experiential learning. In Maurer, F., Ruhe, G., eds.: SEKE. (2004) 19–24
5. Friesen, N.: Three objections to learning objects and e-learning standards. In McGreal, R., ed.: Online Education Using Learning Objects. Falmer Press (2004) 59–70
6. Strother, J.B.: An assessment of the effectiveness of e-learning in corporate training programs. *The International Review of Research in Open and Distance Learning* **3**(1) (2002)
7. Aldrich, C.: Learning by doing: a comprehensive guide to simulations, computer games, and pedagogy in e-learning and other educational experiences. Pfeiffer, Market Street, San Francisco (2005)
8. Webster, J.: Turning work into play: Implications for microcomputer software training. *Journal of Management* **19**(1) (1993) 127–146
9. Stapleton, A.J.: Serious games: Serious opportunities. Paper presented at the Australian Game Developers' Conference, Academic Summit, Melbourne, VIC (2004)
10. M, K., R, L.: Web-based learning: effects on learning process and outcome. *Education, IEEE Transactions on* **45**(4) (2002) 350–356
11. Amory, A., Naicker, K., Vincent, J., Adams, C.: The use of computer games as an educational tool: identification of appropriate game types and game elements. *British Journal of Educational Technology* **30**(4) (1999) 311–321
12. Drappa, A., Ludewig, J.: Simulation in software engineering training. In: ICSE '00: Proceedings of the 22nd international conference on Software engineering, New York, NY, USA, ACM Press (2000) 199–208
13. Sharp, H., Hall, P.: An interactive multimedia software house simulation for post-graduate software engineers. *icse* **00** (2000) 688
14. Navarro, E.O., van der Hoek, A.: Simse: an educational simulation game for teaching the software engineering process. *SIGCSE bulletin* **36**(3) (2004) 233



# The Benefits and Limitations of Self Organizing Feature Map Clustering

John Edwards

Department of Computing Science  
Umeå University, Sweden  
dv06jes@student.umu.se

**Abstract.** This paper presents an analysis of the practical benefits and limitations of the Self-Organizing Feature Map in exploratory data mining. This technique is used to display structures in high-dimensional data starting from heuristics. These structures can reveal correlations and similitudes in arbitrary data sets. Alternative methods of clustering are also discussed and compared. Two basic experiments are provided to illustrate the algorithm and provide performance data for evaluation.

## 1 Introduction

The difficulty of data interpretation varies greatly with data sets and objectives. It is a simple task to calculate the average value of a field from a receipt database containing information of purchases and customers, for instance. However, understanding general relationships between customer profiles and purchased items is a more demanding problem. This is especially true in data exploration phases when one does not know precisely yet what is being looked for. Structural views of the data are then more useful than quantitative analysis. Such graphical views help finding clusters of similar data and this helps defining specific objectives for the subsequent quantitative analysis.

Several data clustering methods can be used to reveal structures in data sets [1]. The present note concentrates on the Self-Organizing Feature Map (SOM) [2]. Because of the interesting features of the SOM, such as combining clustering with projection, the decision to focus on this technique was made. The objective is to elucidate the benefits and limitations of this technique, and this is achieved by processing two illustrative, concrete examples.

The rest of the article is organized as follows. The SOM is defined in Section 2, and the two experiments are detailed in Section 3. The results of other applications of the SOM are then reviewed in Section 4. In Section 5, the performance of the SOM is compared with other techniques. A summary and conclusions are provided in Section 6.

## 2 Background

The Self-Organizing Feature Map, or Self-Organizing Map (SOM), was credited to Teuvo Kohonen [2] in the beginning of the 1980's [3]. In doing so he was inspired by the self-clustering properties of the brains auditory cortex [4].

SOMs belong to the field of artificial neural networks. More specifically to the subfield of unsupervised learning [4]. The SOM provides a method for mapping complex multidimensional input onto a grid of artificial neurons. This grid is usually, but not necessarily, two-dimensional, and rectangular. This is the case that is discussed here. The artificial neurons mentioned are associated with vectors of the same dimension as the input data, weight-vectors, thus rendering the neurons as an abstract concept. The benefit of performing this mapping is that relationships between complex data can be viewed graphically. Data that are close to each other in the input space will also be close to each other on the generated grid [4].

### 2.1 Formal Definition

Using Engelbrecht's notation [4] we denote the neuron or weight vector on the  $i$ th row and  $j$ th column of the grid  $w_{ij}$ . The data to be mapped is presented as an input-vector  $z$  consisting of  $I$ -dimensional vectors, given some  $I$ . Each element of  $z$  is called a pattern. Assume that the grid consists of  $J$  rows and  $K$  columns.

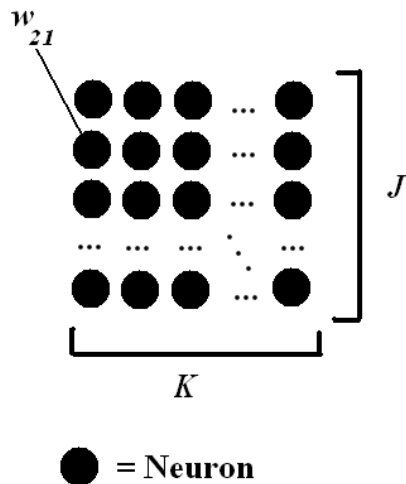


Fig. 1. The SOM.



**The Algorithm** Since the time of its invention, a lot of different versions of the SOM algorithm have been developed. The version described here uses stochastic learning and random weight initialization [4]. Stochastic learning means that the weights are updated after each presented pattern. Following are the steps of the SOM algorithm.

*Initialization* Each weight  $w_{111} \dots w_{JKI}$  is assigned an uniform random value from an interval  $U$ . Ideally  $U$  is formed from the range of the input values.

*Training* The following is done for every pattern  $z_p$  in  $z$ . Let  $w'$  be the new neuron-matrix. Set each neuron to

$$w'_{ij} = w_{ij} + h_{mn,ij} \cdot (z_p w_{ij}), \quad (1)$$

where  $m$  and  $n$  are the row and column indexes of the neuron closest to  $z_p$ . The Euclidean distance measure is used. The neuron closest to  $z_p$  is called the winning node.

The symbol  $h$  denotes the neighbourhood function. A neighbourhood function is a function of two positions on the SOM grid. The function will typically decrease with the distance of these two positions. If the distance between the positions is great enough the function will be zero. Therefore only the neurons close to the winning one will be updated. The distance from the neuron to where the function is zero is referred to as the width of the function.  $h$  is usually a Gaussian function, for instance

$$h_{mn,ij} = n * e^{\frac{-\|(c_{mn} - c_{ij})\|^2}{2\sigma^2}}, \quad (2)$$

where  $c_{mn}$  and  $c_{ij}$  are map coordinates.  $n$  is the learning rate and  $\sigma$  is the width of the function. Note that usually the learning rate and the function width decrease by some factor every epoch. The concept of epochs is discussed below.

*Convergence* The training process is then repeated until some criteria have been met. It is possible to use some error measure to determine when the map is sufficiently accurate. However, it is also common to simply iterate the training process a specific number of times. In this context an iteration is called an epoch.

### 3 Experiments

In order to analyse the SOM two experiments have been conducted. The incentive for doing this is to get a general idea of the strengths and limitations of the SOM that can be compared to other work. It is important to underline the fact that the results of these experiments are not of interest in this paper. These experiments have been provided to illustrate the SOM and to provide performance data for evaluation.

### 3.1 Method

The program SOF-Mapper has been used for both experiments. This program is an implementation, provided by the author of the present paper, of the algorithm described in 2.1. SOF-Mapper creates rectangular 2D SOMs with arbitrary dimensions. These SOMs can then be trained when presented with patterns and a set of parameters. The program can also generate images of the SOM augmented with labels. Each label is associated with a pattern. In the context of SOF-Mapper, a labelled pattern is called a class. Patterns contained in classes do not need to be part of the input-set.

The following process has been used when conducting these experiments:

1. Gather data from the Internet.
2. Select features to gather from the data.
3. Extract the selected features from the data to make patterns.
4. Train a SOM with the patterns using SOF-Mapper.
5. Generate an image of the SOM using SOF-Mapper.
6. Analyse the image.

The specifics of these experiments will be discussed below.

### 3.2 Clustering of Books

The objective of this experiment was to examine how a SOM would map patterns extracted from a set of books onto a grid. The following books were used to extract patterns:

Author	Title	Key
Henry Barbour	The Half-Back	Barbour1
Henry Barbour	The New Boy at Hilltop	Barbour2
Frank Baum	The Lost Princess of Oz	Baum1
Frank Baum	The Enchanted Island of Yew	Baum2
Agatha Christie	The Mysterious Affair at Styles	Christie1
Agatha Christie	Secret Adversary	Christie2
Ian Maclaren	Rabbi Saunderson	Maclaren1
Ian Maclaren	Beside the Bonnie Brier Bush	Maclaren2

Thus eight books, two from each author, were chosen. The books were chosen in a random fashion. The keys are used in figure 2.

Character frequencies were extracted as features to build the patterns. The characters used were ".", ",", ";", ":", " ", "-", "''", "!", "?", and "(". These characters were chosen so that the patterns would reflect the style of writing used by the authors. For instance a high frequency of the "." character would indicate that the author writes short sentences. The frequencies were simply calculated by dividing the number of occurrences of a character by the total number of characters in the book. The following parameters were used to create and train the SOM:

Parameter	Value
Size of the grid	5x5
Dimension of the input-vector	9, the frequency of 9 characters were extracted
Number of patterns	8, one for each book
Epochs	200
Neighbourhood function	Gaussian
Learning rate	0.4
Learning rate decay factor	0.99
Neighbourhood function width	2
Neighbourhood function width decay factor	0.99
Weight randomization distribution	U(0, 1)

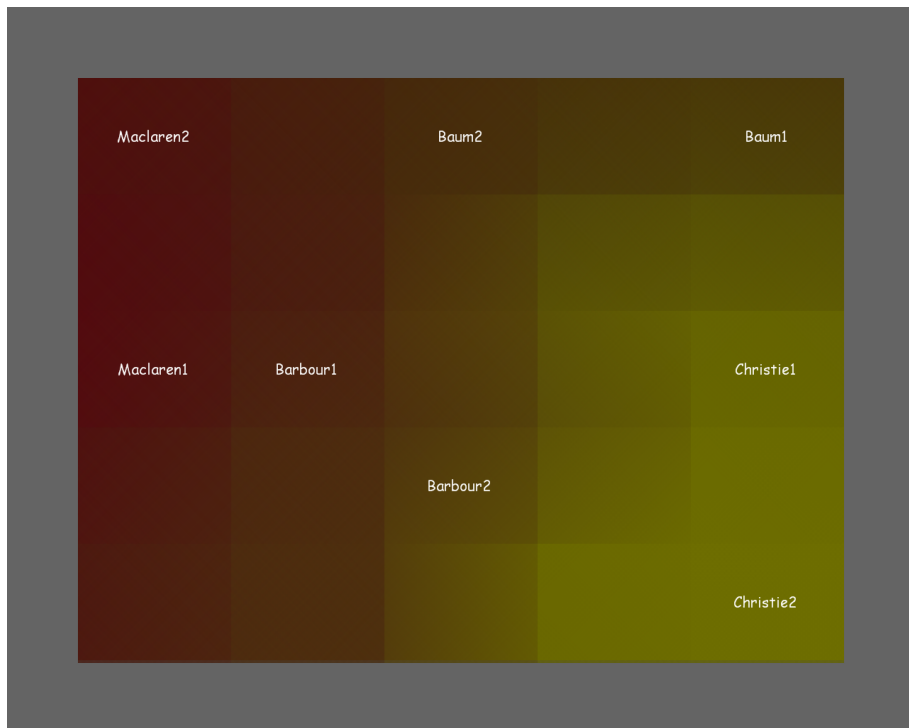
Other sets of parameters were also tested, but most of them gave results that seemed random, inconsistent or singular. The Gaussian neighbourhood function is standard to use. Use of the step function or ramp function gives similar results. Learning was stopped after 200 epochs because the grid was stable by then. Also, the learning rate and function width were very low after 200 iterations, so continuing would not have affected the grid to any greater extent. The size of the grid was set to 5x5 because greater gridsizes resulted in inconsistent grids and lesser gridsizes made the grid too small for SOF-Mapper to clearly visualize.

**Discussion** Looking at the resulting image 2 it would seem that the books are clustered by author. This does not seem strange as it is common knowledge that authors have their own style of writing. However it is important to remember that by simply calculating the Euclidean distance between the patterns we could have concluded that for instance the two Christie books were probably written by the same author. But this does not mean that the SOM is useless. This technique gives us two benefits that we would not get from simple statistical comparison:

- The result is easy to present graphically.
- The entire collection of data is represented on a grid where more complex relationships might transpire.

The image tells us more than what authors wrote which books. Judging by the placement of the Christie patterns and the Maclaren patterns one might suspect that Agatha Christie and Ian Maclaren have the most different writing styles of the four. Of course this is in no way a scientific result of the experiment. However, the image could give hints on where to focus further research.

Based on how the clustering turned out it is not unrealistic to expect that if this experiment was carried out on a full scale with hundreds of books some interesting tendencies might be discovered. For instance the author-clusters might be clustered into super-clusters representing genres. One might also be able to detect which books are written in a different style than usually used by some author.



**Fig. 2.** In this image generated by SOF-Mapper the winning node of each pattern is labelled with the identifier of its associated book as defined above.

### 3.3 Clustering of Politicians

The objective of this experiment was to examine how a SOM would cluster patterns based on Swedish politicians onto a 2D grid. In order to make patterns of politicians the following steps were taken:

1. Text-versions of every official speech and reply uttered in the Swedish public debate building "kammaren" between 2nd October 2006 and 30th March 2007 were downloaded [5].
2. One text-file was created per politician containing everything he or she said.
3. One pattern was created per politician based on the frequencies of certain words used in their associated text-file.

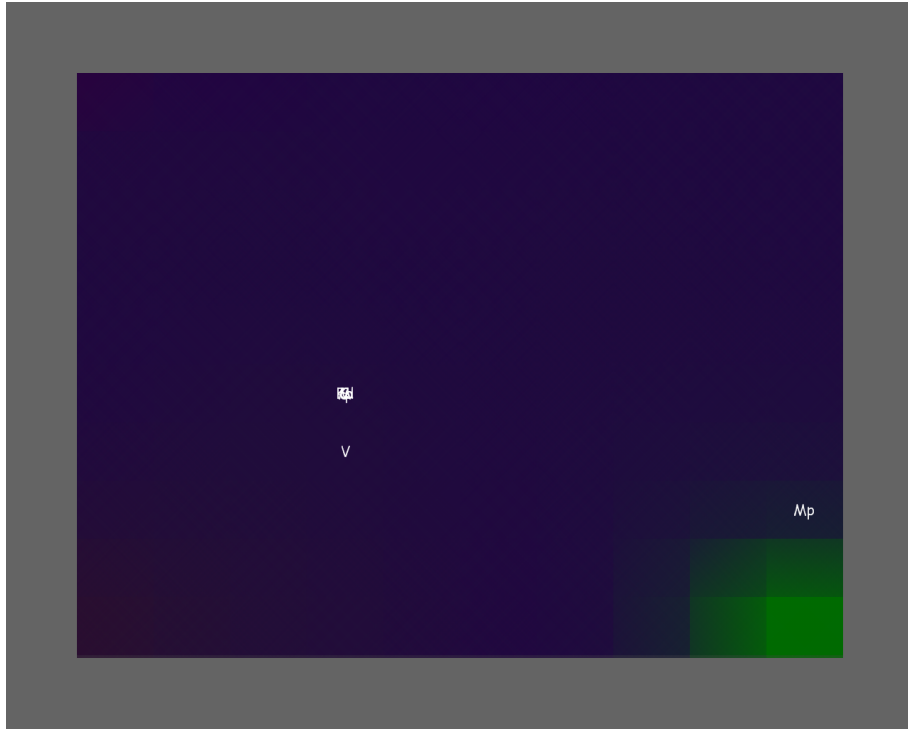
The frequencies of the following words were used to create patterns:

Swedish	English Translation
Arbetslöshet	Unemployment
Miljö	Environment
Regeringen	Parliament/Senate/Government
Ungdom	Youth
Parti	Party
Omsorg	Healthcare
Skola	School
Terror	Terror
Jobb	Job
Pensionär	Senior

These words were chosen because they seem to be used often in political debates, this was verified with frequential analysis. Since the words represent different areas of politics, that different parties value differently, the hypothesis is that they will distinguish the politicians by party. The frequencies were calculated by dividing the number of occurrences of these words by the total number of characters. The following parameters were used to create and train the SOM:

Parameter	Value
Size of the grid	10x10
Dimension of the input-vector	10, the frequency of 10 words were extracted
Number of patterns	356, one for each politician
Epochs	100
Neighbourhood function	Gaussian
Learning rate	0.4
Learning rate decay factor	0.99
Neighbourhood function width	2
Neighbourhood function width decay factor	0.99
Weight randomization distribution	U(0, 1)

These parameters were chosen because of the same reasons as in experiment 3.2. Because there were more patterns in this experiment a larger grid was used. Ideally, the number of neurons in the grid should be no more than the number of statistically independent patterns in the input-set [4].



**Fig. 3.** In this image generated by SOF-Mapper the winning node of each party-class is labelled with the abbreviation of that party. Each party-class is an average of all the patterns of politicians belonging to that party.

**Discussion** In figure 3 all the parties except for Mp are clustered together. It seems as if there are no differences in how members of different parties talk. The exception to this seems to be Mp - which seems very strange at first glance. However only a few Mp-patterns were included in the set, so it is possible that this is just a coincidence. Another possible explanation is that the word "miljö" (environment) was included in the word frequency set. Since Mp is a green party they might use this word a lot more than the other parties. Hence Mp might have a deviate vocabulary.

Note that this does not mean that there are no differences in what politicians talk about. To the contrary, the individual politician patterns were spread out

across the map, they just did not cluster together to form parties. It is possible that they clustered into other classes that were not analysed. For instance members of the same departments might be close together on the map.

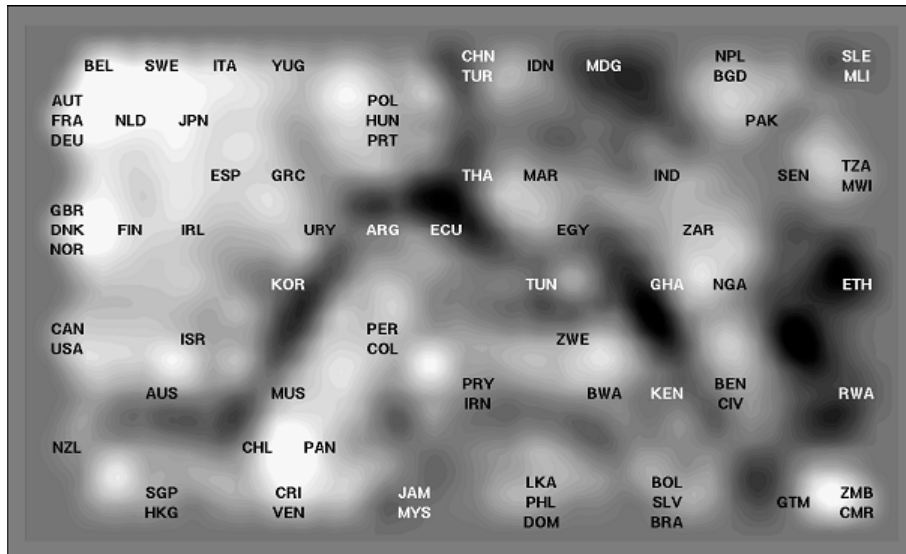
## 4 Related Work

This section will give a brief overview of two other applications of the SOM.

### 4.1 Structures of Welfare and Poverty in the World

In 1996 Kaski and Kohonen published a case study where a SOM was applied to create a map over the countries based on their wealth [6]. In this case study 39 indicators of wealth acquired from the World Development Report were used. In other words, they used one 39-dimensional pattern per country.

The authors of the paper (discussed) point out that the SOM was the logical choice of algorithm. An alternative would have been a method that projects multi-dimensional data onto a 2D-plane. Another alternative was to use one of the classical clustering methods available. However, the authors found the SOM superior in that it combined projection with clustering [6].



**Fig. 4.** This is the resulting image of their case study [6]. As they conclude, the structures present in the image are associated with the countries' GNP and geographical location, even if these measures are not included in the input-patterns. The shades of grey indicate the density of countries. Bright areas are denser than dark areas. (Courtesy of Teuvo Kohonen.)

#### 4.2 Contextual Relations of Words in Grimm Tales, Analyzed by Self-Organizing Map

This experiment [7] was conducted in 1995 by Honkela, Pulkki and Kohonen. As implied by its title the objective was to analyse contextual relationships of words in tales. A SOM was fed with patterns derived from the 150 most common words in the texts, with some exceptions. Each pattern was produced from a word, its predecessor and its successor. Hence expressing the word's context. A rather complex encoding to real numbers was used which will not be discussed here.

As seen in figure 5, the linguistic categories do not form distinct clusters, as one might have hoped for. But in some contexts this could be beneficial. While formal models might provide exact linguistic categories this approach might be more suitable to reflect common language, which is not always grammatically correct [6]. For instance, a SOM could be an important part of a machine translation system [6].

### 5 Other Methods

This section will discuss other methods that are used to cluster data. The popular K-means algorithm and hierarchical clustering will be discussed. The K-means algorithm is a partitional clustering method. Most of the clustering methods used are either partitional or hierarchical, aside from the SOM. There are also methods that project high-dimensional data to low-dimensional data, for instance into 2D, such as Sammon projection [8]. These methods can also be used to overview complex data. However, they will not be discussed here since they do not perform any clustering.

#### 5.1 K-means

The K-means algorithm was introduced by Hartigan in 1995 [9]. Since then much work has been done to improve the algorithm in different ways. The algorithm operates on data that can be viewed as points in multi-dimensional space, that is, the same type of data as the SOM operates on. As described by Vaidya and Clifton [10] the algorithm works as follows:

1. A fixed number of  $K$  clusters is decided. The clusters are sets of vectors, denote them  $C_1 \dots C_K$  and their means  $m_1 \dots m_K$ . Initially each cluster is empty and each mean is equal to zero. Also denote each vector from the input-data  $v_1 \dots v_K$ .
2. Choose  $K$  points from the input-set, denote them  $m'_1 \dots m'_K$ .
3. Set  $m_i$  to  $m'_i$  for every  $i$  in  $[1 \dots K]$ .
4. Put each vector  $v_i$  in the cluster  $C_j$  such that  $m_j$  is of minimal Euclidean distance to  $v_i$  for all  $j$  in  $[1 \dots K]$ .
5. Set  $m'_i$  to the mean of  $C_i$  for every  $i$  in  $[1 \dots K]$ .
6. Repeat 3, 4 and 5 until the mean values are adequately stable.



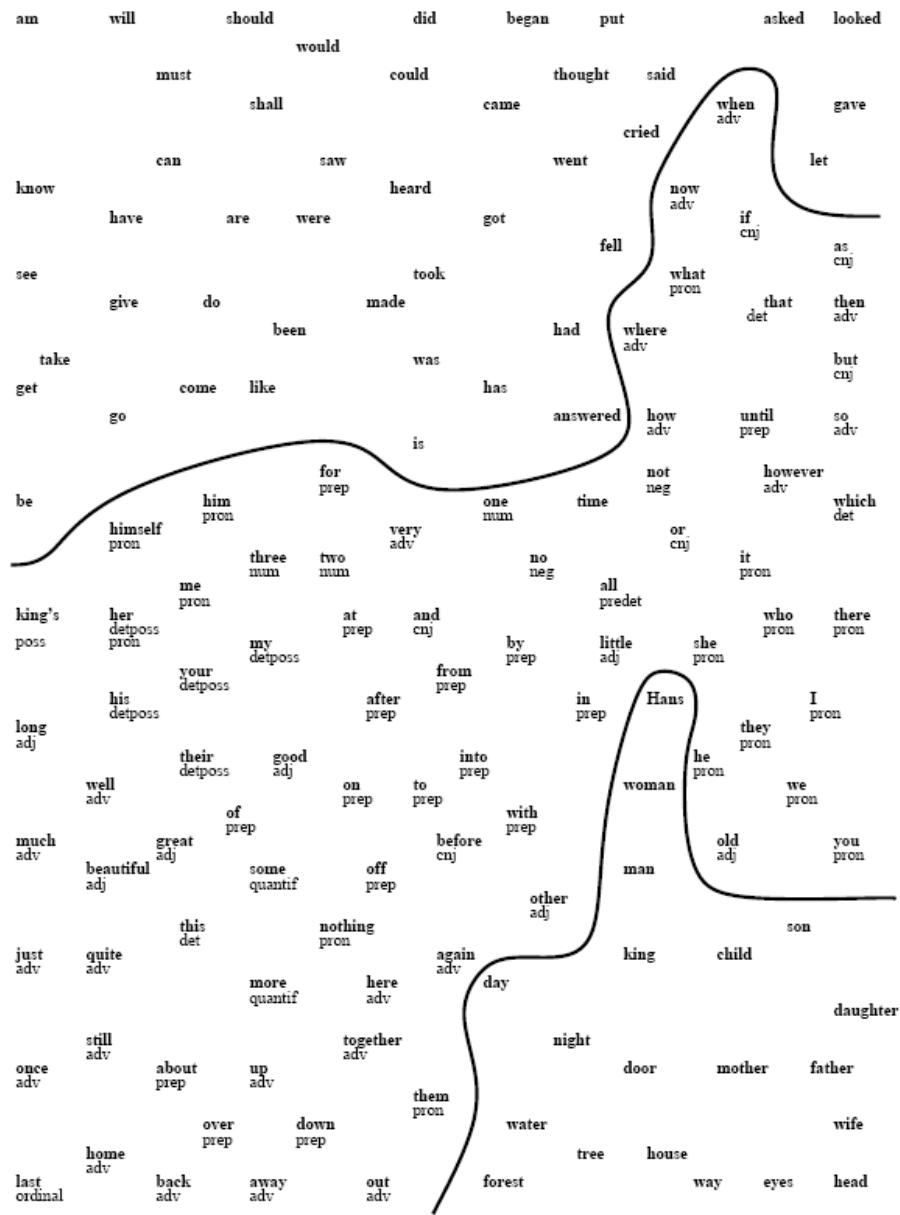


Fig. 5. This image was produced as a result of the experiment [7]. All the words except the nouns and verbs have their associated category printed under them. The verbs and the nouns form two semi-distinct clusters. Other common relationships can also be found at a closer look. (Courtesy of Teuvo Kohonen.)

Steps 3, 4 and 5 constitute the iterative essence of the algorithm. Each cluster is build from all the points that are closest to its centre and then these centres are updated based on the content of their clusters. The algorithm returns both the mean of each cluster and the ownership between each point and cluster.

In a way the output produced by the K-means algorithm is much more general than the output produced by a SOM. Instead of a 2D map we get explicit clusters of the same dimensionality as the input. But there are some major disadvantages with this algorithm compared to the SOM approach.

The K-means algorithm requires us to explicitly define the number of clusters to expect, this number might not be known. However this problem is often dealt with by running the algorithm many times with different values for  $K$  [9]. But this solution does not erase the fact that K-means divides the input into a discrete set of clusters. The structure of the data might be more complex and not simply derived from a set of distinct distributions. Consider for instance data comprised of observations from a dynamic environment, changing slowly with time. It would not be fair to cluster an input-set of this type into distinct clusters, even if the data clearly is structured. Since the SOM does not explicitly form clusters it might be better at dealing with datasets of this type.

On the other hand it might be more comfortable to have the clusters determined by the algorithm. Keep in mind that the SOM does not do this for us unless some post-processing step is included. Ward clustering and the U-matrix method are examples of such post-processing steps [4].

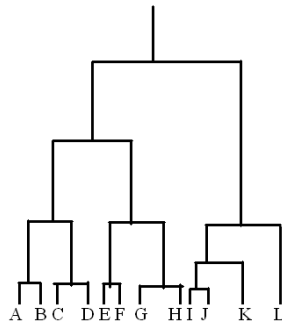
Another difference between the SOM method and the K-means algorithm is that while SOMs project the data down to, usually, 2D-space the K-means algorithm leaves us in input-space. It is not uncommon to have an input-space of very high dimensionality whilst dealing with data clustering. Thus the raw output from the K-means algorithm can be hard to visualize and interpret. However it is of course possible to project the output of K-means down to, for instance, 2D.

## 5.2 Hierarchical Clustering

Hierarchical clustering algorithms generate dendrograms reflecting the structure of the input-set. An example of a dendrogram can be seen in figure 6. As opposed to partitional clustering algorithms such as K-means hierarchical clustering algorithms build hierarchies of clusters rather than a plain set of clusters. The dendrograms produced can be cut at preferred depth to satisfy the sought depth of detail. This set of clustering algorithms is described in detail by Jain, Murty and Flynn [1].

There are many types of hierarchical clustering algorithms. The following description is of the agglomerative single-link clustering algorithm [11]:

1. Create a list  $L$  of Euclidean distances between all distinct pairs of clusters.  
Note that each input-vector is initially considered to be a singleton cluster.
2. Create a cluster consisting of the two closest clusters, according to  $L$ .



**Fig. 6.** This image depicts a sample dendrogram. The letters at the bottom denote patterns.

3. Update the distances and clusters in L. Note that each pattern should only appear in one of the clusters in L. If two clusters are inserted into a new cluster only the new cluster should appear in L. For the single-link version of the algorithm distances between clusters are determined by the minimal distance between two elements in the clusters.
4. Repeat from 2 if there is more than one cluster in L.

There is a very similar adaptation of this algorithm called the complete-link version. The only difference in this algorithm is that the maximum distance is used rather than the minimum in step 3. Each of these versions has some drawbacks and features [1].

An advantage over the K-means algorithm is that the output is more detailed in that it is hierarchical. Since the hierarchy indicates distances among the members of each cluster a visual presentation is not as important. Hence we are not faced with the problem of displaying high-dimensional data. However a grid produced by a SOM contains much more information than a dendrogram.

## 6 Summary and Conclusions

This paper has presented:

- The conduction of two experiments. These experiments were carried out mostly to illustrate the SOM. It was shown that the SOM can cluster data together that can easily be presented graphically.
- Brief presentations of earlier work applying the SOM.
- An introduction to two other clustering methods, namely the K-means algorithm and hierarchical clustering. These alternative methods were also compared to the SOM.

Belonging to the area of artificial neuron science, or soft computing, the SOM has a kind of smoothness to it that does not seem to be common to its alternatives.

The SOM does not group the input-vectors into discrete clusters. Rather a map is presented that is open to human or artificial interpretation. Algorithms are often strict and exact while reality tends to be vaguer. Therefore, in many applications, the impreciseness of the SOM can be viewed as a benefit. In other applications, where exactness is required, it is a limitation.

The projection of high-dimensional data to low-dimensional data is imbedded in the SOM. Since the purpose of the SOM is to reveal structures in data this is an advantage, since high-dimensional data is impossible to view directly. However, if it is required to maintain the dimensionality of the data this is a limitation. The dimensionality could be maintained by using a grid of the same dimensionality as the input. However, for high-dimensional data this approach is infeasible.

In experiment 3.2 the result indicates some similarities between books written by the same authors. But to what extent should the geometrical relationships on the grid be considered? Depending on which features were chosen the outcome differ. The featurevectors are clustered, not the objects they were extracted from. If the authors name was extracted as a feature in experiment 3.2 very distinct clusters would have been formed. If frequencies of English characters were used as features the outcome might have been seemingly random.

There might be some similarity between what politicians of the same party say even if it could not be indicated by experiment 3.3. Word-frequencies do not seem to be the right choice of features, but this does not exclude the possibility of some set of features that would indicate similarity. Perhaps some other set of words or features related to where in the speech the word occurs might have indicated similarity.

A lot of work has been done comparing different cluster techniques. Ultsch claims to have shown that the SOM is the best clustering method available [9]. The SOM has also been shown to be the most trustworthy clustering method used [12]. However both these papers show that hierarchical clustering is better than SOMs in some special cases. It should be noted that these results are based on experiments and should be treated as indications rather than proofs.

One should be cautious about acknowledging a clustering algorithm as "the best one". Different clustering algorithms are good at dealing with different sets of data [9]. Additionally they produce different kinds of output. Thus, a clustering algorithm should be picked based upon the input-set and the desired output.

In practise it is not uncommon to mix different clustering algorithms and methods. For instance the SOM is often used with the U-matrix [9] and the K-means algorithms could be used with some projection technique.

## References

1. Jain, A., Murty, M., Flynn, P.: Data clustering: a review. *ACM Computing Surveys (CSUR)* **31**(3) (1999) 264–323
2. Kohonen, T.: The self-organizing map. *Proceedings of the IEEE* **78**(9) (1990) 1464–1480
3. Deboeck, G.: *Public Domain Versus Commercial Tools For Creating Self-Organizing Maps* (2004)

4. Engelbrecht, A.: Computational Intelligence: An Introduction. J. Wiley & Sons (2002)
5. Linder, L.: Kammarens protokoll (2007) Downloaded 5-march-2007 from <http://www.riksdagen.se/Webbnav/index.aspx?nid=100>.
6. Kaski, S., Kohonen, T., Refenes, A., Abu-Mostafa, Y., Moody, J., Weigend, A.: Exploratory data analysis by the self-organizing map: Structures of welfare and poverty in the world. *Neural Networks* (1996) 498–507
7. Honkela, T., Pulkki, V., Kohonen, T.: Contextual relations of words in Grimm tales analyzed by self-organizing map. *Proceedings of ICANN-95, International Conference on Artificial Neural Networks* **2** (1995) 3–7
8. Sammon Jr, J.: A Nonlinear Mapping for Data Structure Analysis. *Computers, IEEE Transactions on* **100**(18) (1969) 401–409
9. Ultsch, A.: Self organizing neural networks perform different from statistical k-means clustering. *Proc. GfKI, Basel, Suisse* (1995)
10. Vaidya, J., Clifton, C.: Privacy-Preserving K-Means Clustering over Vertically Partitioned Data. *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (2003)
11. Olson, C.: Parallel Algorithms for Hierarchical Clustering. *Parallel Computing* **21**(8) (1995) 1313–1325
12. Kaski, S., Nikkilä, J., Oja, M., Venna, J., Törönen, P., Castrén, E.: Trustworthiness and metrics in visualizing similarity of gene expression. *feedback* (2006)



# Performance Characteristics of String and List Classes in Java 1.6

Timo Elverkemper

Department of Computing Science  
Umeå University, Sweden  
ens04ter@cs.umu.se

**Abstract.** Many performance issues have been discovered in early versions of the Java programming language. The present paper evaluates performance issues concerning Java's string and list classes documented for early versions on the recent release 1.6 of Java and gives guidelines for performance oriented programming with these classes. Our research shows that most issues are still present, but their overall impact on the performance is less significant due to optimizations in the Java compiler and runtime environment.

## 1 Introduction

The performance of the Java programming language has been a heavily discussed topic since its introduction in 1995. Previous work in this area includes performance comparisons of Java to other programming languages like C++, optimizations of the Java Virtual Machine or compiler [1] and also performance limitations discovered during application development [2–4]. These limitations have been documented for earlier versions of Java, but our tests showed that these issues are not generally relevant for release 1.6 of Java. In the present paper we discuss the performance characteristics of Java's string and collection classes in relation to the known limitations in previous versions and give general guidelines for performance oriented programming using these classes. From our experiences in Java development, the string and list classes are a basic building block widely used in application development. Therefore, inefficient or inappropriate use can have a significant performance impact on a Java application. Performance is an especially important topic for the Java language, as programs are run on a Virtual Machine abstracting the underlying hardware platform. This level of abstraction introduces a general performance penalty in comparison to natively compiled languages like C++[1].

**Testing Platform.** The performance testing described in the present paper has been carried out on Windows XP and Linux 2.6.8 to evaluate the performance impact of the underlying operating system and corresponding Java Virtual Machine implementation. But since the test results showed the same magnitude on both operating systems, the runtimes in the present paper are only given for

the test runs on Windows XP. The test programs were run on both an outdated and the most recent version of Java, 1.2.2 and 1.6.0 respectively. Version 1.2.2 was chosen because most of the performance issues discovered in previous work were documented using this or an older version [2–4]. When performing tests with Java 1.6, we used Sun’s “HotSpot Server Virtual Machine” to achieve the shortest runtimes for the testing programs since this Virtual Machine performs more optimizations during execution than the standard Virtual Machine. We also performed test runs using the “HotSpot Client Virtual Machine”, but the resulting runtimes were generally longer than the ones measured using the server version of the Virtual Machine. The results from the client version of the Virtual Machine are therefore not considered in the present paper. The Java version 1.6 is also referred to as version 6, but we will refer to version 1.6 throughout the paper.

**Testing Conditions.** The test results presented in the present paper are average runtimes of 100 timed test runs for each test. Due to the behaviour of Java’s “HotSpot” Just-In-Time(JIT) compiler, it was necessary to perform a full test run prior to running the timed tests. This is needed because the JVM detects the testing code as a “Hotspot” during the first test since most of the CPU-time is spent in the testing code. In turn, this detection then results in Just-In-Time compilation of the testing code which then runs faster and results in shorter runtimes.

Test runs were performed on a Notebook with a Pentium 4 Mobile Processor and 512MB RAM. During the tests only the absolutely necessary processes of the operating system were running. The test programs were invoked with a maximum heap size of 128MB for the Java Virtual Machine.

Task sizes for all test programs were chosen according to two criteria to ensure reasonable test results. First of all, the task size must be large enough in order to be able to measure the runtime of a test. The timer that is accessible within a Java-program is only updated at intervals which depend on the operating system. If a test run is very short, for example around 20ms, the measured runtime is imprecise. Another important point is to chose task sizes so that the test program does not use more memory than is physically available. If the operating system needs to use the virtual memory for the test run, the results will be heavily affected by the paging mechanism of the operating system.

**Structure of the Paper.** The remainder of the paper is structured as follows. Section 2 discusses string classes from a performance point of view. Performance characteristics of Java’s list classes are the topic of Section 3 and Section 4 concludes.



## 2 Java String Classes

### 2.1 Class Overview

This part gives a short description of the classes that handle Strings in the Java language. Unfortunately, class names do not imply characteristics of the actual class implementation. Therefore a basic introduction will be given here. The discussed classes all implement the *CharSequence*-interface which denotes a “readable sequence of char values”[5], which defines methods for reading access to different forms of character sequences.

**String.** Character strings are represented through the *String*-class. All string-literals in the source-code are converted to instances of this class. Internally, the characters are held in an array that exactly fits these enclosed characters. *String*-instances are immutable, all operations aiming to modify a *String* instance need to create a new instance with the desired content.

**StringBuffer.** In contrast to the *String*-class, the *StringBuffer* represents a mutable sequence of characters. The *StringBuffer* manages an array of characters with a predefined size. If this array overflows while adding more characters, the *StringBuffer*-class dynamically extends this array to fit the new characters. This class was implemented with multi-threading in mind. Methods always perform synchronization.

**StringBuilder.** This class is identical to the *StringBuffer*-class with the exception that its methods are not synchronized. Thus only a single thread should work on a *StringBuilder*-instance. The *StringBuilder*-class has been introduced with version 1.5 of Java. In order to perform tests with this class using Java 1.2, we constructed a *StringBuilder* by duplicating the *StringBuffer*-class of Java 1.2 and removing the “synchronized” keyword from its method declarations.

After this description of string classes, we continue by discussing their characteristics from a performance point of view.

### 2.2 Performance of String Classes

The following criteria for string performance mirror the performance issues described in previous articles [2–4]. Each section describes a particular issue, evaluates it according to the current version of Java and gives references.

**Object Allocations.** Looking at Java’s string handling features from a performance point of view leads in the first place to the discussion of string concatenation. As stated in the class description, instances of the *String*-class are

immutable. Thus, modification of a string requires the allocation of a new *String*-instance enclosing the resulting characters and possible garbage collection of the discarded original string.

String concatenation using the “+”-operator is implicitly done using the *StringBuilder*-class as of Java 1.5 while earlier versions use the *StringBuffer*-class instead. As an example, consider the following code sample:

```

1: String a = "string 1";
2: String b = "string 2";
3:
4: String result1 = b + c;
5: String result2 = new StringBuilder().append(b)
6:                                     .append(c)
7:                                     .toString();

```

Interesting in this example is that line 4 is compiled into bytecode that resembles lines 5-7. This behaviour of the Java compiler can be verified using the Java disassembler *javap* and examining the resulting bytecode. Because of this, string concatenation cannot be optimized by the developer through rewriting line 4 to the statement in lines 5-7. It is to note, that the statements in line 4 and lines 5-7 still involve several object allocations. To explain this fact in more detail, consider lines 5-7 of the example code. This statement results in the allocation of four objects on the heap: the intermediate *StringBuilder*-instance, the *String*-instance referenced by “result2” after the statement and two underlying character arrays of these two instances. The allocated *StringBuilder*-instance and its character array are discarded directly after the statement, thus requiring the garbage collector to deallocate the memory. This behaviour has been documented [4] for the outdated version 1.3 of Java and is consistent with the behaviour we found when using version 1.6. This version uses the *StringBuilder*-class instead of the *StringBuffer*-class, but this does not affect the behaviour regarding object allocations.

**Synchronization.** The transition from using a *StringBuffer* to using a *StringBuilder* has obviously been made to address a performance issue documented as “excessive synchronization” [2] or “over-synchronization” [4]. The authors argue that all methods of the *StringBuffer*-class are synchronized. This causes lock acquisition when invoking a method even when the *StringBuffer* is only accessible to one thread. These locks introduce an unnecessary performance overhead in cases where synchronization is not needed. From version 1.5 of Java this issue has been addressed through introducing the *StringBuilder*-class, which is an unsynchronized version of the *StringBuffer*-class.

In order to examine the overhead of using the synchronized *StringBuffer*-class instead of the *StringBuilder*-class when concatenating strings, we measured the runtime of the following code sample:

```

1: int capacity = 10000000;
2: StringBuilder builder = new StringBuilder(capacity);
3: int i = 0;
4:
5: while(i < 10000000) {
6:     builder.append("a");
7:     i++;
8: }

```

For testing the *StringBuffer*-class, line 1 was rewritten accordingly. The internal array of the *StringBuilder*-instance was pre-sized exactly to exclude the effect of expanding this internal array as discussed in the next section. The code excessively concatenates a single character to the *StringBuffer/StringBuilder*-instance. Tests were performed with both Java 1.2 and Java 1.6 in order to examine if the overhead of synchronization described [2, 4] and has been reduced in the newer version of Java. As noted before, the unsynchronized *StringBuilder*-class was introduced in version 1.5. In order to test a *StringBuilder* in version 1.2, we created a *StringBuffer*-class by using the *StringBuffer*-class and removing its synchronization.

The resulting runtimes are shown in Table 1.

**Table 1.** String Concatenation Runtimes

	Java 1.2 (ms)	Java 1.6 (ms)
StringBuffer	1387	404
StringBuilder	608	313

Our test showed that, using Java 1.2, the unsynchronized *StringBuilder*-class only needed approximately 43% of the runtime needed by the synchronized *StringBuffer*-class. But when we ran the test program using version 1.6 of Java, this gap between the *StringBuffer*-class and the *StringBuilder*-class reduced to approximately 22%. Apparently it seems that synchronization is a lot more expensive in Java 1.2 than it is in version 1.6.

**Dynamically expanding Objects.** The *StringBuffer* and *StringBuilder*-classes are implemented in a way that allows the classes to expand their capacity when required. As previously noted, both classes use an array of characters as their underlying datastructure. Arrays are always allocated with a fixed size, expanding can only be achieved by allocating a new array of the desired size and copying the contents of the old array into the new array. In fact, both classes use this method when their internal array has reached its capacity. If  $n$  is the current capacity of the array, then expanding will allocate a new array of size  $2n+1$ .

To summarize, this operation allocates a new array, copies the elements from the old array into the new array and leaves the old array for garbage collection. With increasing array size, this operation will become time consuming. This behaviour has been documented before [3] using Java 1.1.7 and is still valid for release 1.6. In order to evaluate the performance impact of expanding objects and the resulting copy operations, consider the code sample from the previous section.

The test program adds a character to the *StringBuilder*-instance on every iteration. The *StringBuilder* has an internal character array that is allocated according to the constructor parameter. We ran this test program with a value of 1 as the initial array capacity in order to measure the impact of the copy operations and allocation operations that will occur since 10,000,000 characters are appended to the *StringBuilder*-instance. To retrieve a runtime without these copy and allocation operations, the array was pre-sized to 10,000,000 as shown in the code sample when running the test program. The *StringBuilder*-class was used to avoid a performance impact of synchronization on this test. As noted previously, our tests showed that synchronization is more expensive in the older Java version 1.2. Thus, runtime comparisons using a *StringBuffer*-instance are not appropriate here. As in the previous test, we used a self created *StringBuilder*-class because the class library of version 1.2 does not offer such a class.

**Table 2.** Buffer Expansion Runtimes

Initial Capacity (Elements)	Java 1.2 (ms)	Java 1.6 (ms)
1	804	580
10,000,000	581	300

As expected, the copy and allocation operations resulting from the need to expand the internal array of the *StringBuilder*-class is time consuming. The exactly pre-sized *StringBuilder*-instance reduces the runtime of the test program by approximately 30 % for Java version 1.2 and approximately 52 % for Java 1.6. Because of this test results and the implementation details of this class, we conclude that the performance impact of dynamic expansion increases linearly with increasing task size as the number of elements that need to be copied on expansion increases if the *StringBuilder*-instance is not presized to avoid this behaviour.

After the evaluation of common performance issues, we now give guidelines to avoid performance problems in these areas.

### 2.3 Guidelines for optimizing String Performance.

Performance optimized programming with the string classes in Java requires knowledge on both the behaviour of the Java compiler and the actual implementation of these classes. Fortunately, the complete source code of Sun's JDK classes is freely available and provides a good insight into possible reasons for low performance. As for the compiler's behaviour, decompiling the generated class files reveals valuable information.

Concatenation of character strings in a single statement using the "+"-operator as previously discussed is uncritical. The compiler generates byte code that uses a *StringBuilder* implicitly. But when a string is to be constructed over several statements or in a loop, it is essential to use a *StringBuilder* or *StringBuffer* explicitly instead of appending parts of the final string to the *String*-instance using the "+"-operator. Because *String*-instances are immutable, this will result in numerous intermediate objects to be allocated and also immediately discarded for later garbage collection.

When using a *StringBuffer* or *StringBuilder*, it is also important not to use the concatenation operator ("+" ) in the argument to the *append*-method. This will create an intermediate *String* and *StringBuilder*-instance which is discarded directly after the call. In such cases the *append* should simply be called twice in order to achieve the desired result.

Choosing the right class when using a *StringBuilder* or *StringBuffer* is also an important point. Only if the instance is used by more than one thread, the synchronized *StringBuffer*-class should be used. Otherwise the *StringBuilder*-class should be used to avoid unnecessary synchronization overhead.

The performance impact of the dynamic expansion used by the *StringBuffer* and *StringBuilder*-class can be minimized by choosing an initial capacity that matches the final number of characters as closely as possible. But this number can be hard to predict. A too high capacity wastes memory and a too low capacity results in allocations and copy operations. It is the programmer's task to find a good compromise if the final number of characters varies.

## 3 Java List Classes

### 3.1 Class Overview

In order to discuss the performance characteristics and limitations of Java's list classes, we will first briefly describe their main implementation details. The discussed list-classes are part of the Java Collection Framework. The following classes all implement the *List* interface which denotes an "ordered collection" [5] and defines methods to access the underlying datastructure.

**Vector.** This class implements a dynamically growing *Object*-array. If the capacity of the *Vector* is exceeded, its capacity is automatically increased to fit new elements. The initial capacity and the increment that is used on expanding

are programmatically changeable for performance tuning. Internally, elements of a *Vector* are held in an array of fixed size. Expanding the capacity of the *Vector* therefore requires the allocation of a new larger array, copying of elements from the old into the new array and discarding the old array. Methods of this class are synchronized, thus making it usable for several threads at a time.

**Stack.** The implementation of a stack datastructure in Java is provided through the *Stack*-class. It subclasses *Vector* and implements the stack-specific operations like “push” and “pop”. Since the underlying datastructure is identical to the *Vector* class, this class is not discussed any further.

**ArrayList.** The *ArrayList*-class is similar to the *Vector*-class, but does not provide synchronization on its methods.

**LinkedList.** This class implements a doubly-linked list datastructure. The actual data is encapsulated in an inner class providing the linking of the list’s elements. This results in the fact that every data element requires an extra object to be created with it. Methods of this class are not synchronized, but a wrapper class exists for cases where synchronization is required.

### 3.2 Performance of the ArrayList and Vector Classes

The performance aspects of Java’s list classes are similar to the aspects of String classes discussed in the previous section of this article. This results from a similar implementation of the underlying datastructure. Considering the implementations of the classes *StringBuffer* and *Vector* makes the similarity obvious. Both classes use an internal array that is expanded if its capacity is exhausted. Thus, the array-based list classes face the same issue as discussed in the previous section and referred to as “dynamically expanding objects” [3].

Synchronization as discussed in the previous section on string classes is also relevant for the *Vector*-class, but has been addressed by Sun already with the appearance of Java 1.2. In particular, the *ArrayList*-class is an unsynchronized version of the *Vector*-class.

Java’s list classes are unable to store values of primitives types like integers of type `int` [4]. Instead, the list classes can only store Java objects. Primitive Types have to be wrapped into their respective wrapper classes like `Integer`. This introduces an allocation overhead when working with primitive types. From Java 1.5, Sun has introduced a so-called “autoboxing/unboxing” mechanism, that hides the need to wrap primitive types from the developer. However, the mechanism only wraps and unwraps the primitive types automatically and does therefore not change the fact that wrapper objects are allocated.

We implemented a test program that sorts 1,000,000 randomized integer values using the “Mergesort” algorithm [6]. In order to test the performance impact of synchronization, the algorithm was run separately on instances of the

*Vector* and *ArrayList*-class. Additionally, the algorithm was run on an array of the primitive integer type `int` to determine the overhead resulting from the requirement to wrap primitive types into objects. For this test the *LinkedList*-class was not considered since indexed access is far more expensive in a linked list datastructure than in an array. Table 3 shows the results of the test runs.

**Table 3.** Mergesort Runtimes

	Java 1.2 (ms)	Java 1.6 (ms)
Vector	6845	1140
ArrayList	2450	620
Array of Primitives	243	230

Consistent with our findings from the comparison of the *StringBuffer* and *StringBuilder*-class, synchronization had a lower performance impact in Java version 1.6 than in version 1.2. The Mergesort algorithm operating on an *ArrayList*-instance needed between 36 % and 54 % of the runtime needed by the same algorithm operating on a *Vector*-instance. The test showed also shorter runtimes when using Java version 1.6. For the *ArrayList*-class, the runtime dropped by approximately 75 % and for the *Vector*-class it dropped by 83 %. Our implementation of the “Mergesort” algorithm uses only two methods of the *ArrayList* and *Vector*- classes, namely “get” and “set” for indexed access. Since the implementations of these methods are unchanged and the implementation of our test program on Java 1.2 and 1.6 is identical, we conclude that optimizations made in the Java compiler and runtime environment lead to the improved performance of version 1.6.

The test runs based on an array of primitive values showed that the convenience of using the *ArrayList*-class comes with a high performance penalty. Using Java 1.2, the Mergesort algorithm operating on the array of primitive values needed only approximately 10 % of the time needed when using an *ArrayList*-instance. For version 1.6, the performance gap is smaller, but using the array of primitives requires still only 37 % of the runtime needed by the Mergesort algorithm operating on an *ArrayList*-instance.

We also performed test runs with different sizes of input for our testing program that implements the “Mergesort” algorithm. The results from this tests all showed very similar percental differences in runtime between using the array of primitives and the *ArrayList* and *Vector*-classes. Because of this, we conclude that the actual runtime difference in percentage between the different datastructures is independent from the size of input for the “Mergesort” algorithm.

### 3.3 Performance of the *LinkedList* Class

In contrast to the other list classes, the *LinkedList*-class is neither synchronized nor based on an array as its internal datastructure. Therefore, the previously discussed issues referred to as “dynamically expanding objects” [3] and “excessive synchronization” [2, 4] are not applicable to this class. But as for all list classes, primitives need to be wrapped into their respective wrapper class for use in a *LinkedList*-instance. In addition to this, the *LinkedList* introduces another allocation overhead since every element in the list is wrapped into an extra object that provides the actual linking of the list. In comparison to an *ArrayList*-instance of the same size, this results in allocation of twice the number of objects and finally also leads to increased activity of the garbage collector.

Extensive testing has been done for several Java versions up to 1.4 to compare the *LinkedList* and *ArrayList*-class from a performance point of view [7]. We used the testing program developed by Jack Shirazan [7] to evaluate the performance of insert operations and iteration over the elements of a list for version 1.6 of Java. The performance of insert operations was tested by inserting elements at the beginning, middle and end of the list.

**Table 4.** Insertion of 10.000 Elements into an *ArrayList* and a *LinkedList*

	Java 1.2 (ms)	Java 1.6 (ms)
<i>ArrayList</i>		
Beginning	7952	7681
Middle	4163	3915
End	271	60
<i>LinkedList</i>		
Beginning	351	130
Middle	32927	36933
End	340	121

The results in Table 4 show that the worst case for the *ArrayList*-class is insertion of an element at the beginning of the list since all elements have to be moved. This is the only case where the *LinkedList*-class provides better performance. For the other cases, inserting at the middle and end of the list, the *ArrayList*-class needed less time than the *LinkedList*-class.

Surprisingly, the test results for insertion at the middle of a *LinkedList*-instance showed actually lower performance on Java 1.6, while all other tests were faster on Java 1.6.

A testing program iterating over the elements of both an *ArrayList*-instance and a *LinkedList*-instance with 1,000,000 elements each shows also better performance on an *ArrayList*-instance for Java version 1.6. Iterating over the *ArrayList*-instance required only 29 % of the time needed for iterating over the *LinkedList*-



instance. For the older Java version 1.2, the *LinkedList* was slightly faster in our test runs. The detailed results for the test are shown in Table 5.

**Table 5.** Iterating over an *ArrayList* and a *LinkedList* with 1,000,000 Elements

	Java 1.2 (ms)	Java 1.6 (ms)
<i>ArrayList</i>	66	15
<i>LinkedList</i>	57	51

After discussing the performance characteristics of the *ArrayList*- and *LinkedList*-class for some of the standard operations on these classes, we continue by giving guidelines for performance optimized programming using these classes.

### 3.4 Performance Guidelines for optimizing Performance of the List Classes

Performance oriented programming using the list classes discussed in the present paper requires knowledge on the classes' implementation details. Generally, the *ArrayList*-class showed the best performance in most of the test runs we performed. There are however special cases where a *LinkedList*-class should be used. We agree with Jack Shirazan [7], that the *LinkedList*-class should be considered if elements are frequently inserted at the beginning of the list. Additionally, when elements matching a given criteria are to be removed from the list, the *LinkedList*-class should be used, since an *ArrayList* would need to move elements whenever an element is deleted. The *Vector*-class should only be used if synchronization is required, otherwise the synchronization introduces an unnecessary overhead. From a strict performance point of view, an array should be used instead for a list class whenever possible. Especially working with primitive types is much faster using a primitive array as shown in our Mergesort example.

## 4 Conclusions

Most of our testing programs show that the performance of Java's string and list classes has increased significantly from Java release 1.2 to 1.6. Since the testing programs are implemented identically for both releases, we conclude that optimizations in the Java compiler, Virtual Machine and class library lead to this improvements. Performance issues in version 1.2 that are related to synchronization have been addressed through the introduction of new classes. Other performance issues like "dynamically expanding objects" are still present, but our tests showed a less significant performance impact when the tests were performed with Java 1.6. Performance oriented programming with the string and

list classes requires a good knowledge on how these classes are actually implemented. It is important to choose the correct class for a specific task to achieve optimal performance.

## References

1. Schatzmann, J.C., Donehower, H.R.: High-performance java software development. *Java Report* **6**(2) (2002) 24–41
2. Heydon, A., Najork, M.: Performance limitations of the java core libraries. In: *JAVA '99: Proceedings of the ACM 1999 conference on Java Grande*, New York, NY, USA, ACM Press (1999) 35–41
3. Klemm, R.: Practical guidelines for boosting java server performance. In: *JAVA '99: Proceedings of the ACM 1999 conference on Java Grande*, New York, NY, USA, ACM Press (1999) 25–34
4. Shah, M.A., Madden, S., Franklin, M.J., Hellerstein, J.M.: Java support for data-intensive systems: Experiences building the telegraph dataflow system. *SIGMOD Record* **30**(4) (2001) 103–114
5. Sun Microsystems: Java platform, standard edition 6 api specification (2007) <http://java.sun.com/javase/6/docs/api/>, accessed 2007-05-22.
6. Goodrich, M.T., Tamassia, R.: *Data Structures and Algorithms in Java*. 2nd edn. John Wiley & Sons (2001)
7. Shirazi, J.: *Java Performance Tuning*. 2nd edn. O'Reilly (2003)

# Designing for Real-Time Collaboration Through Small Screens

Reza Assareh

Department of Computing Science  
Umeå University, Sweden  
dit03rah@cs.umu.se

**Abstract.** This work investigates the need of a real time collaboration application on mobile hand devices and how today's internet infrastructure supports this type of collaboration model. It presents brief design guidelines of how to design for Real-time collaboration (RTC) through small screens and scenarios in which the need of such RTC application is demonstrated. It also presents a prototype application interface for the two types of mobile hand devices that are most common in the market today, the ordinary mobile phone and the PDA's. The results described here are the suggested prototype and an investigation of the needs of a RTC application on a mobile hand device. The findings highlight the usefulness of real time collaboration on mobile hand devices and the need for additional features to support collaboration across representations.

## 1 Introduction

Over the past few years the way of meeting and collaborating have developed a lot. Dramatic efforts to the upend 20th-century model of collaboration have been done, among them, increased accountability and charter collaboration. Computer-mediated communications (CMC) along with Real-time collaboration (RTC) is recognised as a medium of learning and collaborating that is highly interactive and capable of supporting interaction as well as collaboration between people. [1]. Both these technologies involve Internet communication and are referred to as audio and video conferencing, instant messaging, web conferencing, whiteboarding & screen sharing, document sharing and groupware. The traditional computing environment require the users to connect to each other through a wired computer and may be ineffective or inefficient in many situations. Requiring mobility, the solution was to make computers small enough that they were easy to carry or even to wear, such as mobile devices. The ability to communicate and collaborate any time and from anywhere gives both the IT software vendors and Telecom organizations opportunities to strategically take advantage of the RTC methodology for increasing productivity, speed, and customer service. No doubt the mobile connection has had a profound effect on our lives, our work and play, our politics, and our business. But, in the middle of this revolution that seems so profound, no one is yet quite certain what the landscape for such mobile RTC environments will look like. Social networking

through RTC in a professional context, building and sharing knowledge and collaborating around a task are considered as problems because of the size of the small screens.

The main purpose with this paper is to present some benefits of the RTC concept as well as the opportunities for applying the concept on mobile hand devices. Further this paper examine a designer's approach toward the development of environments for RTC through small-screen mediators and devices such as the mobile phones or the PDA's. This is critical because the final product must be effective for both the technical and the usability part within the virtual environment. How can the designer ensure effectiveness, and what should he/she focus on? What strategies could guide the designer through out the process of developing online information and how can we evaluate the result?

## 2 Background

### 2.1 Computer-supported co-operative work (CSCW)

The concept for supporting collaborative activities and their coordination through computer systems were first developed by Irene Greif and Paul M. Cashman in 1984 [2] and is referred to as *Computer-supported co-operative work* (CSCW). CSCW, also called groupware, borrows interaction metaphors from formal human meetings like the telephone and video conferences to build its own supporting infrastructure for collaborative activities [3].

Within CSCW extensive attention has been paid to technology that supports remote collaboration between individuals. The development of these collaboration systems and technologies, have started debates concerning the possible changing quality of organisations and organisational activities [4]. For example, mobile technologies, whether these are mobile telephones or more sophisticated devices, allow organisational activities to not be within a particular desk or a stationary computer.

### 2.2 Computer mediated collaboration (CMC)

As the same time as the CSCW concept was born, leading scientists tried to develop systems that allow people to communicate with each other when they are in physically different places. Main concerns in this concept have been how to allow people to communicate as if they were in the same room [5]. Collaborative technologies such as email, videoconferencing, computer conferencing, chat rooms and messaging are well known examples. Other less commercial technologies as collaborative virtual meeting places, such as *Second Life*, are getting more and more common. Collectively, all these communicative technologies supporting collaboration are referred to as *Computer mediated communication* (CMC) [1] and in combination with systems that allows the user to take part of collaborate activities, they are together referred to as *Real time collaboration* (RTC).

### 2.3 Real-time collaboration

A collaborative system or technology can be characterized as a system where a group of users have come together with the intent to exchange (and share) data, state transitions and actions initiated by participants. The data shared could be text, graphics, shared computer displays or multimedia content.

Real time collaboration systems allows people to work together at the same time, even when some or all participants and their work products are in different physical locations. To do this effectively, the software must support a way of giving participants enough cues about each other to help them organize their interactions. [6]. Consequently, the design and implementation of a distributed system supporting real time collaboration must handle the human factors of how people collaborate as well as the expected technical issues.

One of the fundamental problems within collaborative systems is how to disseminate the right content to the right participants. Real-time collaboration use often Internet technology to communicate with other co-workers as if they were in the same room. Furthermore, since the participants in a collaborative session are distributed over a wide area of network, supporting their collaboration needs to handle complications that can occur during communications, network failures and group memberships.

Within the Real-time collaboration concept, several kinds of synchronous communication tools are involved.

- **Instant messaging and Group chat services** Tools to exchange immediate messages with connected buddies. This also includes an enhancement to Instant Messaging where multiple users can exchange information while still being able to see previous messages.
- **Friend list and awareness technology** The user maintains a list of people that he/she can interact with and also be able to see the other users current status mode, online, offline, busy or away.
- **Whiteboard collaboration** The electronic equivalent of a blackboard, but shared between the users. Whiteboard systems allows the participants to view one or more users writing on an on-screen blackboard.
- **Application and desktop sharing** The user is able to share applications, documents, or desktop to arrange online meetings.
- **Voice over IP** Two-way audio transmission.
- **Video and audio conferencing tools**
- **Annotation tools**

### 2.4 Benefits of RTC (and mobility)

Besides the many benefits of the RTC concept such as mobility and that individuals can work collaboratively despite geographical dispersion. Recent terrorist threats and the economic situation are encouraging many companies to take a closer look at these benefits to avoid unnecessary travel, decrease overall expenses and lower cost of ownership, and increase productivity. Increasing bandwidth and

cheaper hardware [7] makes the concept and its tools more attractive by the day. There's no doubt that the value of Web-based mobile real-time interactivity is tremendous, especially where responses are needed immediately and the decision makers are distributed around the globe.

Mobility is a feature of something that can be identified as a moving and a flexible item. Mobility in the RTC concept does not only imply that the platform where the application is running on can be moved [3], it also gives the concept a more flexible characteristic. The user is no longer bound to use the application on a stationary computer but can now reach the other co-workers and the project from anywhere. It also allows people to keep abreast of the goings-on in the organisation without having to leave their home offices.

Using RTC enables lots of collaboration options both for individuals and the teams working with the same session through a real time collaborative communication tool. However, because of many of these options are not integrated into the familiar tools used in the daily work environment, they are sometimes not part of a cost-effective solution. To integrate the RTC concept into a mobile platform will partly overcome this problem but the design of such solution need to consider the mobile device's limitation in screen space and interaction methods.

### 3 Market observations

Currently there is very little research in the field of RTC on mobile hand devices. The classical RTC application is used often on a stationary computer connected to the internet. These RTC solutions consist often of a windows-based application combined with the tools (that was earlier mentioned) necessary for the RTC concept (see figure 1). To suggest a alternative way of interaction using the RTC concept, I will take a look at existing techniques that uses the RTC concept. Here, I have studied three solutions, Instant Messaging (Skype, MSN), Web 2.0 and Second Life.

#### 3.1 Instant messaging

*Instant messaging* is considered as "the grand father" of all real-time collaboration applications. IM has its roots in *IRC* (Internet Relay Chat) and dates back to 1988 [2]. IM has become the preferred means of communication through internet and its users have accounts on multiple IM systems. IM is still predominantly a text-based chat application though most public and enterprise IM services now support live voice and video calls. IM became popular due to the fact that it was the least invasive way of communicating with someone. It doesn't interrupt people as much as phone calls and it doesn't require constant attention. One other cause is that the conversations can go on for long period of time. For example, users keep a running IM conversation with other colleagues and co-workers during a long time as they collaborate with a shared project. IM is incredibly helpful, not only does it have a constantly open channel for communications, but also the log of previous messages. Finally, due to the concept of presence,

it allows individuals to see who's on-line before initiating conversation, or using their own presence status to keep interruptions to a minimum. For project team members, awareness of presence is invaluable, maybe more important than e-mail. As we're moving further into the 21st century, vendors are integrating their Internet-based chat (IM) applications with Web concepts, and extending the functionality of IM services [8].

### 3.2 Web 2.0

There are several web applications made especially for the RTC concept. They all have very similar design and interface and are made to be used through the web. The applications use the traditional tools needed for the RTC concept and are often a paid service. For example, one commercial application named *ConceptShare*, is using a typically flash-application made to be used through the web.

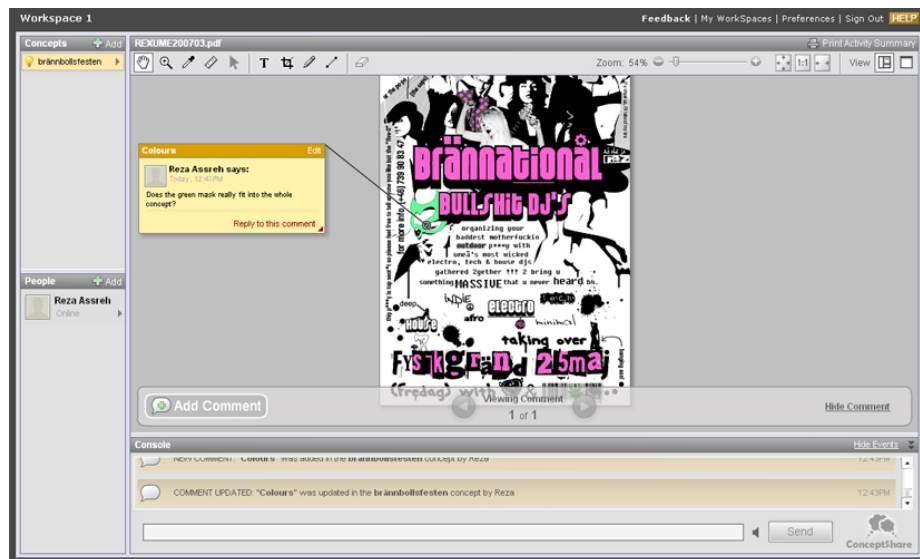


Fig. 1. A web solution for RTC, *ConceptShare*. [www.conceptshare.com]

### 3.3 Second life

Slightly surprising but frequently used solutions for the RTC concept are *virtual workplaces* such as *Second Life*. *Second Life* is mainly considered as an entertainment vehicle, but it is also bringing alternative ways of how groups of individuals collaborate. Some enterprises are actually using *Second Life* as their primary collaboration environment.

Second Life is a 3-dimensional virtual reality environment in which residents create avatars and use a software application to explore a virtual world. In the last few years the Second Life growth has surged, with over 4 million registered accounts [9]. With a growing number of users and an open source programming code, individuals are discovering ways to make real-life money selling goods and services within Second Life. Vendors have recently carried out several Second Life initiatives as well, such as allowing virtual attendance to conferences and the recent launch of a Second Life developer community for collaboration on development of 3-D virtual applications [9].

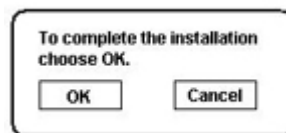
As a collaboration tool, Second Life offers a richer user experience, enabling users to move from the 2-D world of web conferencing into a virtual world that can enable direct user-to-user communications. For example, suppose that the user is in a virtual meeting room watching a presentation on the screen. The user can tap another participant on the shoulder and ask them a question. The user is also enjoying a far richer user interface than simple web conferencing can currently provide. Second Life has significant potential but must overcome perception problems that will cause distraction and doesn't offer any business value. Predicting organisations have promised that over the next ten years Second Life and other virtual reality enterprise environments will transform the Internet. [9]

## 4 Interface studies

The available screen space of a mobile device is almost always smaller than the amount of data to be represented. Due to the lack of space and the large amount of data that need to be represented techniques have emerged to increase the display size by virtual design means. These techniques include design of various windows and dialog boxes as well as zooming and panning techniques.

### 4.1 Dialog boxes

Dialog boxes are used to take priority over anything in the background. These are best described as a pop-up box that appears in various linear user-guidance situations and are used as interface elements to offer the user clear, step by step guidance.



**Fig. 2.** An example of a dialog box.



## 4.2 Windows

Application windows are used to present information in. The screen size of today's mobile phone are approximately between 128-200 x 128-176 pixels. By increasing the resolution, bigger amounts of data can simultaneously be visible on a small display. For displays of same physical size, a high resolution results in smaller rendering of the graphical elements, making it possible to present more data comparing the the low resolution screen [7]. Windows got dynamic properties such as free moving and sizing but despite that they are not used on small screens since it requires the use of two hands.

## 4.3 Tabs

For organizing user options, one of the most frequently used design elements on small screens is tabs. Tabs are similar to an index-card system which organizes the user's available options in an easy and quick way [7]. Tabs are often used as an alternative to pull-down and pop-up menus but the designer should have in mind that having more than five to seven tabs displayed at the same time will confuse the user.[7].

## 4.4 Pull-down and pop-up menus

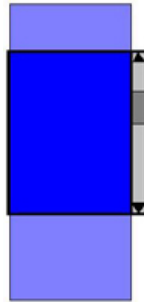
When it comes to menu structure, pull-down and pop-up menus are techniques that offer selection of options available for the user. The pull-down structure, drops down the menu from the top edge of the screen where as pop-up menu structure opens up from the bottom edge of the screen. These kinds of menus are frequently used for the quick selection of the fields in forms and are common on PDA mobile application.

## 4.5 Panning and leafing

To pan large areas within a window a common solution is to use scroll bars. The bar works both as an interactive control and a dial. Sometimes the bar is a proportional bar to indicate the user's current position in relation to the total content in of the page as shown in figure 3.

The proportional slide also shows the proportion of the visible content in relation to the total content of the page which is very useful to the user's knowledge about how much information the page contains [7]. On small screens scroll bars are used for one direction- vertical, whole page navigation. Vertical navigation also lends more easily to the single-handed, thumb-based operations which mobile phones often tends to use.

Leafing is the second technique that offers an easier way to navigate in a limited space. The content of a window or a screen can be split into portions and be presented in several pages like traditional book formatting [7]. On application which is designed with a stylus, scrolling is more complicated than leafing however on mobile phones a single handed scrolling operation seems to be the better choice.



**Fig. 3.** Shows an example of a scroll bar.

#### 4.6 Zoom

The zooming technique is yet rarely used in mobile phone applications though it brings excellent functionality. The technique allows large amounts of data to be presented in a small space. One of the reasons why the technique is rarely used is that zooming is a demanding technique, both for the user (calls for a higher cognitive attention) and the mobile phone (so far the mobile phone screens have not reached the resolution and processor demands) [7]. The ultimate zooming technique for small screens is free zooming within an application that allows continuous or free selection of the focus point (disproportional zooming) [7].

#### 4.7 Dividing small screens

For an interface designer one of the most important decisions to make, is to decide how the screen should be divided. Here the designer's choice of layout for the content is crucial and is different depending upon what type of applications he/she is designing the interface for [10].

Vertical formats need a static navigation area situated in the short side of the screen. This will apply to the text based application and provides maximum line length for reading [7].



**Fig. 4.** An example of a vertical screen when divided into two sectors of interaction and presentation.

For horizontal formats, the arrangement of the content will come easily but the problem here is that only few lines of text can be displayed at the same time. This screen dividing format is requiring the user to scroll even for short texts so therefore a vertical dividing format in combination with a static navigation area on the short side of the screen is recommended.



**Fig. 5.** An example of a horizontal screen when divided into two sectors of interaction and presentation.

#### 4.8 Physical interaction techniques and elements

Small screens are often meant for mobile phones and beside the interface design, a multitude of different input and interaction elements have been developed for the operation of small mobile devices. Interacting with mobile phones is different from computers in various ways such as how the application presents feedback, demands for a one- or two-handed interaction or if the user interaction is held through a touchscreen, mini joystick, click wheel, Anoto pen, voice input or with traditional keys [7]. Most mobile phones and PDA's supports most of these interaction techniques but depending on what application that is running, different techniques (often in combination) are used for the physical interaction. The collaboration approach that is suggested here is focused to be used with traditional interaction techniques available on the most common mobile phones and PDAs.

#### 4.9 Sharing mechanisms

When designing collaboration and coordination for mechanism it is important to consider how socially acceptable the system is to the user. If the system is not acceptable the users will not use the system in the way it was intended or simply abandon it [1]. A key issue, is to get the right balance between human coordination and system coordination. If the system has too much controls then the user will rebel. If the user control too much, the system will collapse. One example is the case with file sharing and locking. The common solution here is that the sharing application uses file locking [1]. To prevent the users from clashing when trying to work on the same part of a shared file or document,

whenever someone is working with a file, it becomes inaccessible to others. This is not a good way of solving the problem, because when collaborating on a shared document it is essential that there is more than one user at the same time. A more flexible way of solving this problem is to include a social policy of floor control [1]. Whenever a user is working on a part of a document he/she must initially request "the floor". If no one else is using the specified section of file/document that time then the user is allowed access to edit that part. Some applications such as *Google document* is using a combination of floor control and fast saving. Fast saving is the mechanism of saving and updating the document in small time intervals so everyone that is editing a shared document is aware of what the latest updates are.

## 5 Design guidelines

Applications that are developed traditional RTC tasks are often complex and extensive. When designing for interaction, it is often hard to create an explicit link between the physical form, the interaction concept and the software application. Because of this, it's important to create links that follows our mental model and allow us to see the "invisible" space inside the device. Though the mental model doesn't need to correspond to the actual structure of the software, this type of presentation enables meaningful links and logical interrelationships between the real world and the virtual world [1]. The solution presented here uses the practical form of a metaphor as mental model. The metaphors form a narrative framework in which the possibilities of the system can be placed in a context that is logical for the user. While working within a application that can handle a large amount of projects and files, an important factor when it comes to orientation is the preservation of context [7]. The RTC application suggested here have a great focus on the user orientation, using techniques that stretch or compress the content for long lists or large tables such as calendars and gives the user a good overview of the possible functions as well as the actual project.

When it comes to organising information there are five different categories that can be used to sort the information by location, alphabetical order, time (sequential), category (context) and hierarchy. On computer-aided systems, all of these can be used in combination to organise the information [7].

On small-screen devices, a special challenge for the designer is the presentation of text. The designer has to develop a concept of legibility that works in the worst possible condition. For example, will the lighting condition be ambient or will the user be in a screen location to concentrate fully on the screen. The most important factor is the contrast and brightness level. The brightness contrast should be at least 50% in order to ensure good legibility for display of text on small screens [7]. Since navigation menus and elements are using very small text, a font that is specifically developed for small screens should be used. The antialiasing technique uses halftone pixels to visually smooth the edges of a character but should not be applied to a type that is smaller than 12 pt as the text will be blurred and difficult to read [7].

On small screen, different highlighting methods are more suited to use for indicating headings, links or instructions. Recommended is to use bold characters as long as they don't run into each other. The use of colour is suitable for highlighting text while using italic type is generally unsuitable for small screen displays [7].

When designing for small screen the use of icons is important, popular and justifying. An icon is often logically connected by the narrative framework and allows non-verbal communication between the user and the system. Even though icons work well across language barriers, the designer should avoid confusing the user and achieve visual consistency. Icons within a system should be designed to have the same degree of abstraction [7].

In many cases the same application is used in different ways depending on what platform the application is used on. For example, studies have shown that using email services differ if the user is using a mobile device or a computer. On mobile devices, users only check and read new incoming mails, the mobile device works here only as a presentation device because the users very rarely choose to answer their emails right away. Instead the users wait till there's a computer available later [11]. For a designer it is important to consider a mobile device for such RTC application mainly as a presentation tool. The main focus should be on representing the information as easy as possible and second to use the functions as smoothly as possible.

## 6 Prototype

During the design process, in order to develop an application that takes needs of potential users in focus it is helpful to use potential user scenarios. The scenario methods lists the requirements for an application fairly precisely and gives guidelines for realistic interaction situation. Here there are three typical scenarios for some ordinary RTC procedures through the user's mobile device.

### – Case one

Hans is a part of a design team as a graphic designer. On their last project the team needs to gather for a last meeting before handing over the finished advertising product. Hans lives outside the city and this morning all the transportation to the city is closed due to an accident and thereby he can't make it in time to the meeting. He decides meet with the rest of his design team through his mobile phone.

### – Case two

Martin and Johanna are working together on a layout project for a website. After a couple of days work with the layout of the webpage, Martin discovers that there are some ideas that they really should try out. Since Martin and Johanna are living in different places, he contact Johanna by the telephone to describe the new ideas but Johanna does not really understand the new ideas. To help her understand he uploads his suggestions on their common RTC application so Johanna easily can follow his ideas even though she's not in the office.

– **Case three)**

Caroline has had a lot to do lately and is forced to stay in after closing time to finish her writings on a scientific article on LaTeX. When she's done with her article and is about to make the references, she's having trouble with getting the code right. Since the teacher is home, she starts a workspace with the code she's using now and invites her teacher to take a look at it. The teacher spots the problem and corrects her code within a few minutes so Caroline can deliver her article just in time.

To understand how easy this kind of RTC interaction is, an outline of a subsequent prototyping activity is shown below. The menu structure is the same for both the mobile phones and PDA application.

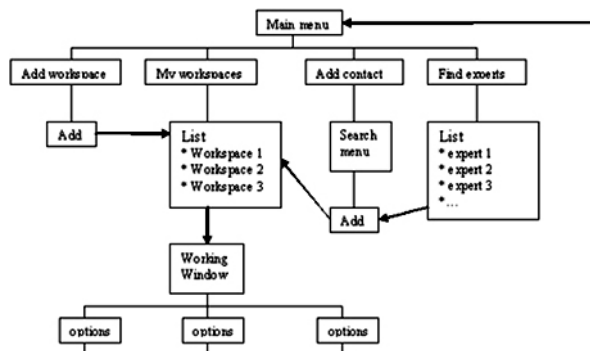


Fig. 6. The prototype proposes functionality with a roadmap as shown here.

From the main menu, the user can choose between four options, to add a new workspace (project), view and work on previous projects, add a new member to the project (co-worker) or find a co-worker from an external expertise (this service is optional).

When adding a new workspace the user will see the current project among the previous projects, here the user can choose which project to work on. In the working window the current project with its members included open to discuss and work on, here the main tools of the RTC concept is located such as chat, file sharing and annotation. After saving the user is back to the main menu.

## 7 Conclusion

When designing for any application, the planning and development of such interactive collaboration service that is presented here is very complex. To accomplish a good design and user interface, the design process must be iterative, meaning that every suggested hypothesis must first be discussed and evaluated between



**Fig. 7.** The first screen when using the RTC application on a mobile phone device. The user can choose between the application four main functions, adding new member, adding new workspace, work on previously projects or find experts to invite into their projects.

several disciplines and then repeatedly modified. Also the interaction steps must be planned with caution, the designer has to design so that the user is liable to be intolerant and distracted when using the product.

This study explains the benefits of computer-supported co-operative work as well as the potential market of mobile computing and the usage of RTC in mobile devices. Further more we describe how these elements together enable a network structure that supports the utility of the RTC concept on small screen devices. Further this work presents mock-ups and an early state screenshots from a prototype that uses multiple design disciplines to achieve the requirements of a stereotype user. The design guidelines presented here address the problem of supporting RTC concept on small screens such as lack of screen space, diverse interaction and different user needs. Focusing on realistic task, the prototype allows the members involved in a collaborative project to reach their work through a mobile platform from almost anywhere. It demonstrates collaborators choosing mobile hand devices to reach their work online and highlights the need for such RTC technique. The scenarios exemplifies how a successful project depends both on collaborative teamwork and on real-time services which meet the challenges of a distributed work force from any location. With teams no longer based in a single location, collaboration cannot happen exclusively in face-to-face meetings or in coffee corners.

## References

1. Jennifer Preece, Helen Sharp, Y.R.: Interaction Design, beyond Human-computer interaction. John Wiley & Sons, Ink, Reading, MA (2002)



**Fig. 8.** When adding new workspace, the user can name and give every workspace an unique discription.



**Fig. 9.** The new added workspace will appear in the list of current projects.

2. Wikipedia: Wikipedia homepage (2007) [Http://www.wikipedia.org](http://www.wikipedia.org), accessed 2007-03-01.
3. Gulliksson, H.: Design of Mobile Applications. Umeå Universitet, Reading, MA (2006)
4. Bellotti, V., Bly, S.: Walking away from the desktop computer: Distributed collaboration and mobility in a product design team, Proceedings of CSCW '96 (Cambridge, MA), ACM Press, 209-218 (1996)
5. Strategies to Increase Interaction in Online Social Learning Environments. (2000)
6. H., C.: Using language. Cambrigde University Press ISBN 0-521-56745-9 n/a(1) (1996)
7. Schmitz, C.Z..B., 7.5, S.: Designing for Small Screens. Ava Publishing, Reading, MA (2005)
8. Lazar, E.: Im 2.0 (2007) [Http://www.collaborationloop.com](http://www.collaborationloop.com), accessed 2007-04-19.





Fig. 10. In the working menu, the content of the project is presented, further the user can choose additional tools to edit, comment or chat discuss the project.

9. Insider, S.L.: Second life insider homepage (2007) [Http://www.secondlifeinsider.com](http://www.secondlifeinsider.com), accessed 2007-04-21.
10. Saffer, D.: Designing for Interaction, Creating Smart Applications and Clever Devices. New Riders, Reading, MA (2007)
11. Jacobsen, O.: Mobil specialbehandling. Metro Teknik 2007 n/a(25 April) (2007)



Fig. 11. In the tools menu, the available tools for RTC are presented to choose between.

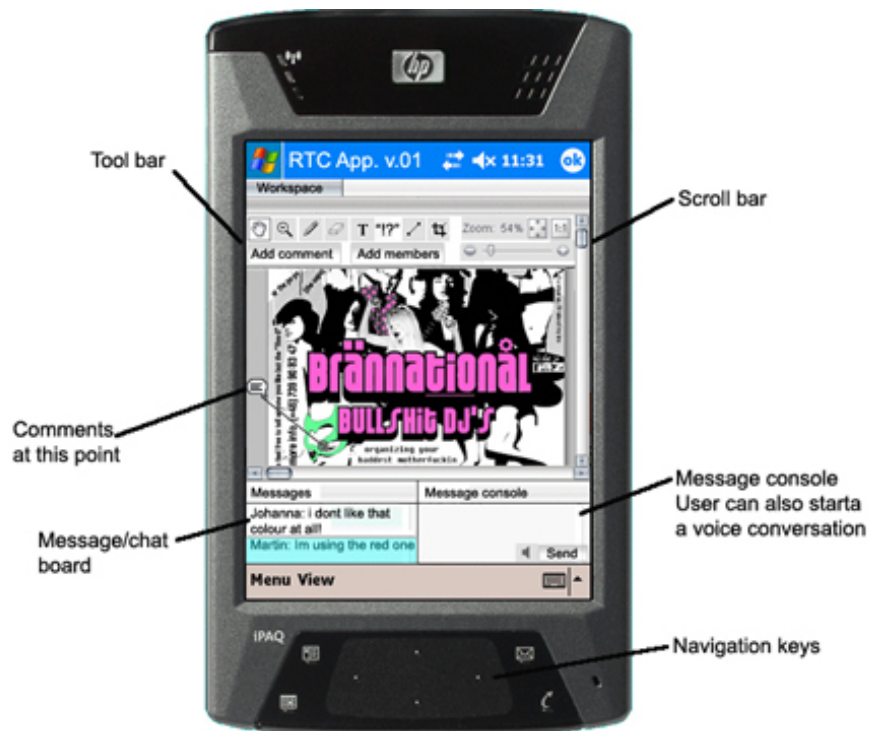


Fig. 12. The figure shows an interface prototype made for PDA's. Since PDA's got touchscreens and a larger screen, the interaction is made through a slightly different way.

# Designing emergent interaction: Adaptive interfaces with emergent behaviours

Jakop Berg

Department of Computing Science  
Umeå University, Sweden  
ens02jbg@cs.umu.se

**Abstract.** An adaptive interface requires that a system can interpret the users' preferences from the information given to the system's user-model. Therefore the designer must predict the future users' goals and actions before the user can interact with the finished system. If a system can redesign itself, not only by the current user's preferences but also according to preferences of other users with similar needs, then its design becomes dynamic and can emerge as the users interact with it. This article examines how systems can, by the use of an adaptive interface with emergent behavior have the ability to adapt to unexpected changes in its users' preferences. It gives an example of such a system and possible benefits from it. And discusses, what issue needs to be considered when designing it.

## 1 Introduction

The growth of the internet, the World Wide Web, and an increased general use of computers create an environment where a wide variety of users with different backgrounds, interests, experiences, skills and learning styles access the same applications. They use computers for purposes ranging from personal entertainment to collaborative work in critical projects [1].

This diversity puts high demands on the application interfaces. A good interface has to support different groups of users, with different types of backgrounds and needs. There are significant differences in users' experiences of an interface depending on his/her groups' cognitive style. Different groups of users approach tasks in different ways. For example, when coming in contact with a system for the first time one group of users will start to study the instruction manual and another group will immediately start using the system and turn to the manual only when in trouble [2].

Web applications and systems connected to networks have great possibilities to collect large amount of information about users and their interaction with the system. This kind of information can be used to create adaptive interfaces that support all kinds of users. This article will discuss systems where the interface adapts not only to the current user but also according to other users with, for example, similar user preferences and cognitive style. This article will also talk about whether these kinds of systems can enhance the user experience by

using emergent behaviors. The article will do this by covering some relevant subparts that creates such a system. The remainder of this paper will start with describing the basics and benefits of adaptive interfaces and provide an overview of user-modeling (section 1.2, 1.3). In Section 2 the article addresses emergence, emergent interaction and emergent interfaces can exist. Section 3 and 4 cover design issues that need to be considered when constructing emergent interaction and adaptive interfaces. An example of how such a system and the possible benefits from it is given in section 5. Finally in section 6 I will summarize my conclusions about what I will call emergent adaptive interfaces systems (EAIS) and issues involved in designing them.

### 1.1 Adaptive interfaces

The idea to construct user interfaces that are suitable for all users regardless of their knowledge is a basic principle of human computer interaction (HCI). To achieve this it is necessary to make the interface suitable for any user at his/hers given level of knowledge at that given time. This is sometimes called universal access, which is based on “good user-based design” and has the goal to address the needs of all potential users on their conditions [3].

Making interfaces that suite everybody creates needs for systems and interfaces that can adapt to the context where it’s being used. To achieve this adaption there are two main techniques, user-invoked adoption and automatic adoption. User-invoked adoption is a commonly used technique, where the user can choose between different modes [3]. The customization is done through preference choices which make adjustments of domain specific functionalities, size and proportion of interactive elements [3, 4]. For example the user can choose between simple or advanced mode. Where an experienced user requests complete menus and short prompts, the novice user rather uses short menus and more informative prompts [4]. In automatic adoption the system adapts by the user letting the system collect information about him/her and in this way predicts on what level the user is or which function he/she is likely to use [3]. Different methods of user modeling are used to achieve automatic adoption. The focus of this paper will be on the second type of user-adoption, i.e. automatic adoption. More about user modeling will be covered in section 2.3.

When constructing interfaces and systems, an adaptive user interface is an important tool to simplify complex tasks. It can be used to limit the amount of information displayed while removing information that is not relevant for the current user. This method can be especially helpful in the growing amounts of different mobile applications and devices where screen size limits the amount of information that can fit on the screen simultaneously. It is proven in Sears and Schneiderman study of the split menu system that Adaptive interfaces can increase the efficiency and the users’ experience. However, because the menus were constant for each user in their study, it does not take in consideration the problems that can arise with constant changes of menu structures [5]. These effects and how to deal with them will be discussed later in section 3.1 of this report.

Jameson gives, makes in his article “Adaptive Interfaces and Agent” [5], three points to why the use of adaptive interfaces is something that will continue to grow:

1. *Diversity of contexts and users.* Computing devices are being used by more and more people, and in more diverse contexts. To design system that’s will be suitable for all users in all context without using some kind of user-adaptivity will become harder and harder.
2. *Complexity and number of interactive systems.* The number and complexity of the systems that users will have to deal with continues to increase. Therefore helping users to deal with interactive systems even if they lack the will or knowledge to understand the system, will become increasingly important.
3. *Increasing scope of information to deal with.* Today’s users can often access more diverse and larger amounts of information than a few years ago. It is therefore getting more attractive to delegate some of that work to a system. Even if it has a imperfect model of the users requirements.

## 1.2 User modeling

To be able to make an interface that adapts automatically to its user/users, the user/users have to be modelled in some way, the system must predict the user’s wants and needs. I will therefore briefly explain some basics of user-modeling. The field of user modeling is going back over 20 years and is by now a mature research area with well-established techniques that are empirically evaluated [1].

A user model contains all of the information that the system knows about each of its users. The user model usually starts as a default set up and maintains and develops by the system on the basis of events and the user’s actions in his/her interaction with the system. This interactions might be mouse clicks, completion of tasks or requests for assistance. Therefore the user model will be in constant change and will gradually adapt to the user’s development as he/she increases his/her knowledge of the system. The model can contain anything from novice/expert distinction, layout preferences, preferred device and interaction technique, to cognitive oriented models about what the user knows about commands, how he/she maps tasks onto sequences of commands or the user’s view of an application concept. The system uses rules to interpret its information about the user. With help of the interpreted information the system can adapt to the user’s needs, abilities and knowledge [4].

To be able to identify the needs and the preferred preferences of the user, stereotypes of different kinds of users are often used. A designer can by categorizing users determine what parts of a system the user is most likely to use. By categorizing users the designer can simplify the design process and reduce computational load at run time. More sophisticated systems supports multiple stereotypes that even might be contradictive [1]. There are different methods and techniques to use in order to collect the information the system needs to predict what the user wants and create a user model. Gerhard Fischer mentions some of these in his article “User Modeling in Human-Computer Interaction” [6]:

- Collecting information by the user telling the system
- Collecting information by inferring what the user wants from the users actions or user data.
- Collecting information by external events communicating information to the system.

There are a wide variety of analysis techniques and user models to support different kind of applications. Bill Klues mentions some typical attributes, inputs and techniques maintained in user modeling. These are showed in table 1 below:

**Table 1.** Typical attributes, inputs and techniques maintained in user modeling.

Attributes typically used in user models.	Input methods to the user model.	Techniques for creating, deriving facts and analysing form user profiles.
<ul style="list-style-type: none"> <li>– User preferences, interests, attitudes and goals</li> <li>– Proficiencies (e.g. task domain knowledge, proficiency with system)</li> <li>– Interaction history (e.g., interface features used, tasks performed/in progress, goals attempted/achieved, number of requests for help)</li> <li>– User classification (stereotype)</li> </ul> <p>Specific values for the attributes may be explicitly derived by the analysis engine, captured directly from user actions, specified by the user.</p>	<ul style="list-style-type: none"> <li>– Explicit preferences, goals from questionnaires</li> <li>– Explicit personal characteristics (e.g., job title, level of education)</li> <li>– Self assessments</li> <li>– Specific actions</li> <li>– Vision and gaze tracking</li> </ul>	<ul style="list-style-type: none"> <li>– Bayesian (probabilistic)</li> <li>– Logic-based (e.g., inference techniques or algorithms)</li> <li>– Machine learning techniques (e.g., neural networks)</li> <li>– Stereotype-based</li> <li>– Inference rules</li> </ul> <p>The user model permits the current knowledge of the user to be combined with the domain, task or other models to derive new facts.</p>

Another way to model user preferences that Anthony Jameson [5] mentions in “user modeling by Collaborative filtering”. By collecting the user’s opinion about objects the system can predict a user’s opinion about another object that he/she has not yet rated. By comparing what users with similar preferences as the current user has had about this object. Collaborative filtering is used by

many web based systems as a way to model the preferences of user. And Studies has shown that collaborative filtering techniques are able to make relevant recommendations to individual users with usefully high accuracy.

As more and more information is collected and stored by different companies and organizations, the danger that this information will be misused increases [6]. The privacy issue of user modeling is something we need to be aware of and take in consideration when designing systems that might be intrusive to the user's privacy. I will not go further into the privacy issue. Even if this is a very important aspect of this kind of systems it is outside the scope of this paper.

## 2 Emergent interaction

**Emergence.** When a system grows in complexity, it is not unusual that the number of subsystems and the effects they have on each other increase. This can often lead to a system where the global behavior can be difficult to predict by studying its subsystems alone. In other words, the result of interaction between the subsystems, which might be the phenomenon of interest, gets difficult to predict. These behaviors can in some cases be considered as emergent. The expression emergent describes behaviors that arise when subparts interacts in complex systems with enough complexity, natural or artificial [7]. Emergent systems are categorized by that it's difficult to predict their global behavior by studying its subparts, and that it is difficult to determine which subpart that creates a desirable global behavior [8]. Other characteristics for emergent behaviors are:

- The subparts that are involved in creating the global behavior are distributed.
- The global behavior arises from the individual actions of each participating subpart
- Emergent behaviors can occur on any level, but always in a bottom-up way.

Despite the fields popularity and the amount of research done in the subject, scientists has yet to agree on exactly what emergence are, and its final definition. There is no way to separate systems or a model of a system that really are emergent from the ones that only shows emergent behaviors. It's therefore a risk that that the term emergence is used to describe systems where there is a lack of theoretical knowledge to describe it in a better and more correct way.

**Emergent Interaction.** According to Andersson et al. emergent interaction is a field inspired by a new kind of applications where the interaction between individual and the collective is a common theme. In focus are the social aspects of the application. Some of these applications have emerged and developed in different direction that was intended by the product developers from the beginning, by individual use of the technique. Emergent interaction systems (EIS) are described as:

*“Emergent interaction system is an environment with a number of actors who share some experience/phenomenon, and whose behavior is significantly influenced by a shared feedback loop picking up data from the individuals and their actions” [7, p 14].*

Significant in these systems are the impact the user’s actions have on the system, and how the feedback generated by the system effects the individuals and the collective. Something emerges from the interaction between the shared phenomenon, the collective and the individuals as a result of the feedback mechanism. This will hopefully enhance the users’ experience of the system and the shared phenomenon. EIS are always situated in a context. This fact makes an EIS able to be part of another EIS. Which makes it possible for an EIS to be defined by which other EIS that are up and running and participating in the broader system. The architecture of Emergent interaction systems is characterized by more or less centralized autonomous units that have the ability to interact with other units in its environment [7].

**Emergent adaptive interfaces.** A kind of interaction systems where the adoption of the interface is effected not only by the single user, but by a collection of all user models, could in some cases be considered as EIS. The system is its own environment and the interface is the users shared phenomenon/experience and the collection of user models serve as the feedback loop. In this kind of system every user works by their own preferences in his/her part of the system, maybe even unaware of some or all of the other parts of the system. This will create a situation where the user has no way of knowing exactly what affects his/hers actions might generate for other users or the global environment of the system. A situation like this might be in some aspects comparable to ant-algorithms, which are often used to simulate emergent behaviors. These simulations are made by having agents called ants that work by simple rules without knowing the global aim of the system [8]. In many emergent systems sensors are used to detect changes in the environment to be able to collect data about the phenomenon for the feedback loop. In a system as the one presented above the sensors role in the system is replaced by the user’s interaction with the system and the user model created by it.

### 3 Designing adaptive interfaces

In contrast to other types of interactive systems, adaptive interfaces requires, as we have seen earlier in this article, that the system has knowledge of its users, careful designed methods for collecting information about their users, sophisticated computational techniques to interpret what the user’s goal actually is [5], and an interface that’s suitable for adoption [4]. But even after fulfilling this, it might be hard to empirically prove that user-adaptively has benefited the system [5]. So constructing a successful adaptive interface is not an easy task,



it requires careful user studies and preparations. Gerhard Fischer [6] mentions some of the issues that are needed to address:

- When and how to assist/interrupt the user. If this is not done right, the adoption will do more harm than good.
- What is the problem domain, which are the possible actions and what are reasonable goals for the use. To be able to assist a user, the system has to know his/her goals.
- How and if to construct user classification. There are many different kinds of users, with different experiences and usage patterns. That makes normal classifications by stereotypes insufficient for some types of systems.

Most systems are complex enough for a user to have good knowledge about certain aspects of the system but not another. One expert user might use one part of a system but not another, in cases like this it is suitable to create stereotype users based on the type of tasks they perform rather than of expert/novice user knowledge [5].

No user adaptive system (UAS) can be designed on principles alone. No matter how sophisticated techniques are used, empirical work and studies have to be done in order to ensure an UAS stays in touch with reality [5]. These systems are often far too complex for a designer to be able to predict their behaviors without some kind of empiric testing.

One of the bigger challenges in information-rich systems is not to make all information available but to make the right relevant information available at the right time presented in the right way. This creates a fundamental problem of UAS software design. At design time design an application that works as it was designed especially for the individual user when he/she uses it. Even if his/hers preferences are not known until the very moment he/she uses it. Designers have to try to predict the user's patterns, preferences and context before the user even sees the product. By using user-modeling and adaptive interfaces, the difference of design time and use time gets blurred. When the system constantly adapts to the user, it can take advantages of contextual factors. Use time becomes a different kind of design time as the system redesigns its self as a consequence of the adjustments of the users model [6].

A solution to the growing diversity of usage context, users and devices, seems to be adaptive user interfaces, but there is also a downside that needs to be taken in consideration [9]. "Analysis of context-aware user interfaces shows that adoption mechanisms have a cost-benefit trade-off for usability" [9, p 301] unpredictable changes or incomprehensive adoptions of the user interface can easily decrease a systems usability. This is especially critical for elements very frequently used in the interface [5]. These downsides might be one reason that successful adaptive interfaces are hard to find, and that adaptive interfaces have a tendency to appear unpredictable, incomprehensible and give the user a feeling of losing control. It is therefore important that the interface is designed with the right support for the user. So he/she can understand the changes in the interface made by an adaptive interface [9]. It has been showed that the negative aspects of adapting user interfaces are possible to avoid by using the right design. It can

be done without changing or improving the users' mental model. By giving the user real-time information, telling him/her on what grounds adoptions in the interface are made the system can support its decisions for the user. This information will make the user more aware of the underlying reasons for adoptions and give a better understanding of them [9]. Finally the system should always leave the final decision of adoption to the user. By doing so the user will feel more in control of the system and the changes will not feel arbitrary [5].

Bill Kules [1] presents in his article "User Modeling for Adaptive and Adaptable Software systems" some guidelines for designing adaptive interfaces.

1. *Know thy user.* Adaptive user interfaces requires not only that the system designer knows the user, that knowledge has to be embedded in to the system and the system needs to be able to act on that knowledge.
2. *Don't forsake good HCI principles.* Adaptive system techniques are based on the some foundation that HCI are. Without these the adaptive interface will not have any benefits for the user.
3. *User centered design.* Make your design from with the user in focus. Start evaluating you design early and often. Cost of failing to completely understand the user requirements and characteristics can be higher than in traditional user interface development.
4. *Don't destroy the users' sense of control.* If the user don't understand way the interface changes and for what reasons. The changes will appear arbitrary and unjustified. This will reduce their sense of control. Always present the user with information of the user model and changes, give them a choice if he to accept or decline all changes. Make it possible to adjust attribute values at any time.
5. *Support the users' experience.* When adopting the interface make sure that the user don't need to abandon achieved skills, strategies or mental models.
6. *Expect, test and support challenges.* An adaptive interface will increase the complexity of the system. Testing techniques must be adapted to the different states the system can take. The support desk will need to be able to determine the state of the interface when identifying the user's problem. You can simplify this task by making the user model accessible and adjustable.

## 4 Designing emergent interaction

Since emergence in some way is an unexpected property, is it really possible to design for emergence or does it become something else the very second it is designed? In that case, there would of course be no point in discussing design of emergent interaction. But what can be done, is designing for the possibility of emergence. Even if the unpredictability is inherited from the concept of emergent system we might be able to design for emergence, to provide an environment that are supportive for emergent behaviors, and that encourage it [7].

According to Andersson et al, advanced systems like the ones discussed in this article where the emergence is a part of the system being developed, the

designers can be viewed as online actors in the system. The designers are trying to steer the emergence in what they believe to be the right direction. At the same time taking advantages of lucky but unforeseen effects that might emerge, and with the help of these changing their view on what the right direction is based on the systems actual development in time. Emergent design has in contrast to traditional system design, its focus on what effects and purpose the application has, and works by an iterative redesign that is guided by emergent system analysis. They propose that the methodology for such projects might be like this.

- Determine on a hi-level, what results the EIS should have.
- Design for emergence by designing a system that has possibly the necessary characteristics to satisfy some identifiable requirements for desired effects. And that might with some luck produce these requirements.
- Test your design by simulations or prototypes.
- Evaluate and compare how close you come to your system goals.
- If the goals are not satisfied. Try to analyze what part might be the reasons for the results, and start over. If the goals are partly achieved analyze the emergent system and try to identify what parameters and design decisions have lead to which effects. Then redesign, evaluate and iterate until you're system has achieved its goals.

## 5 Schematic example of a system containing an adaptive interface with emergent behaviors

The concept of EAIS can seem abstract and complex to comprehend. This section will therefore give a schematic example of how an EAIS could work and what the benefits would be in this example. By constructing a system with the following features:

- New functions and functionalities can be added to the system by plug-ins written by the users themselves or by others.
- The system models its users and is able to adapt its interface and functionalities to the user's needs, skill level, cognitive style and what type of tasks he/she most frequently performs.
- The users can customise not only the functions of the system, but also their accessibility according to the user's preferences (e.g. the position of buttons, shortcuts, menus, active corners etc.).
- The user rates the goodness or the importance he/she thinks the different features and functionalities have.
- By comparing users with the help of collaborative filtering, the system can propose new features and changes that should suite the individual user.

It could be possible to have a system where the individual user's actions and customisation of the system affects the global behaviour of the system. The benefits should be that the system will suggest new features and functionalities

to the user based on how other users with similar needs have applied them. The system will at the same time adapt to the type of tasks the user performs. Further more as the user progresses and develops, the system will adapt to the users new knowledge. Updates of the system can be released without the need for new releases by any company. Users with knowledge in programming can develop plug-ins and updates themselves when the needs for new features are identified. Plug-ins can in collaboration with each other create unforeseen functionalities that were originally not intended by the developers. Good solutions and features will be used by more users than inferior solutions, which will allow them to spread to other users with similar needs and benefit others in their use and interaction of the system.

## 6 Conclusions

By constructing a system that adopts both to some aspects of the current user and the collection of other users' user-models it is possible to create a system with emergent behavior in its interface and this might benefit the individual. What the final design will look like in such a system is hard to predict since it constantly changes. The users are unconsciously redesigning the interface by their use and adoption of the system. User time has become design time which hopefully can produce interfaces that better respond to the users needs. The system can identify and suggest functionalities that benefit the user. This can help users to find new functions and ways to perform tasks that they might not have found or realized themselves.

**Design of emergent adaptive interfaces.** To be able to successfully develop a system with an emergent adaptive interface it is likely that the designer needs to consider design issues that are common when designing both adaptive interfaces and emergent interaction systems. When designing the part of the system that models the user's behavior it is of course important to consider all issues normally involved in such systems and the same should be true for designing the adaptive interface part of the system. But because of the extra complexity in designing emergent adaptive interface systems (EAIS) I believe some of the issues and design principles listed in the article are more important than others. One reason is that in the case of a successful EAIS, as described above, the emergence should take care of filtering out bad solutions and unnecessary functionalities while giving support for good ones. I have in the list below summarized the issues in this article that I believe are the most relevant when designing systems with these characteristics. I have also tried to give some motivation as to why they are important.

Issues important to consider when designing EIAS:

1. *Inform the user.* Make the user understand how and why the interface changes, so he/she does not lose the sense of control. Leave the final decision to the user. If the user does not understand or accept the changes,

there's a big risk that they will not refrain from use them. This can lead to that a good solution does not get used and does not spread as intended.

2. *Design for timing.* Do not interrupt the user when it is not necessary, and only display information relevant to the user. If the user is not informed in a suitable manner, he/she will get irritated, not embrace the systems features and turn them off. In which case the whole idea of the system will fail.
3. *Get to know the system users.* Learn to know the users of the system and be sure to evaluate what information will be relevant for the user-model, and in what way you need to interpret that information. It is important that the user-model correlates with the users. Otherwise the adoption and suggestions will not benefit the users and they will not use them.

Design methods:

1. *Put the end-user in focus.* Use end-user centered design. Perform empiric studies of both your system and its users. Make frequent evaluations, and expect the cost for testing to be higher than when developing traditional systems. It is difficult to predict what the effects will be of a system like this, that lies in its nature. If not tested and evaluated thoroughly, it is very hard to predict its behavior.
2. *Iterate.* The design process is iterative. When designing complex systems the designer should expect many iterations and redesigns. The complexity of EIAS makes them extra difficult to design and perform in the desired way. A system that only almost works is not good enough since the users' acceptance of the system are so important for it to work properly.
3. *Be receptive.* Embrace new ideas and solutions to problems that might emerge in the design process. Use them as inspiration. If you try to control the results too firmly you will miss the point in using emergent design. The benefits lie in the unexpected.

## References

1. Kules, B.: User modeling for adaptive and adaptable software systems. In: Proceedings AMC Conference on Universal Usability, Arlington, USA (2000)
2. Chen, S.Y., Magoulas, G.D., Dimakopoulos, D.: A flexible interface design for web directories to accommodate different cognitive styles: Research articles. *J. Am. Soc. Inf. Sci. Technol.* **56**(1) (2005) 70–83
3. Stephanidis, C.: Adaptive techniques for universal access. *User Modeling and User-Adapted Interaction* **11**(1-2) (2001) 159–179
4. Sukaviriya, P.N., Foley, J.D.: Supporting adaptive interfaces in a knowledge-based user interface environment. In: *IUI '93: Proceedings of the 1st international conference on Intelligent user interfaces*, New York, NY, USA, ACM Press (1993) 107–113
5. Jameson, A.: Adaptive interfaces and agents. In Jacko, J.A., Sears, A., eds.: *Human-Computer Interaction Handbook*. Erlbaum, Mahwah, NJ, USA (2003) 305–330 Available from <http://dfki.de/~jameson/abs/Jameson03Handbook.html>.
6. Fischer, G.: User modeling in human-computer interaction. *User Modeling and User-Adapted Interaction* **11**(1-2) (2001) 65–86

7. Andersson, N., Broberg, A., Bränberg, A., Janlert, L.E., Johansson, E., Holmlund, K., Pettersson, J.: Emergent Interaction a pre-study. UCIT, Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden (2002)
8. Flake, G.W.: The computational beauty of nature. MIT Press, Cambridge, MA, USA (1998)
9. Paymans, T.F., Lindenberg, J., Neerinx, M.: Usability trade-offs for adaptive user interfaces: ease of use and learnability. In: IUI '04: Proceedings of the 9th international conference on Intelligent user interfaces, New York, NY, USA, ACM Press (2004) 301–303

# Mobile phone interfaces for the visually impaired

## – A study

Fredrik Björnskiöld

Department of Computing Science  
Umeå University, Sweden  
bjornskiold@gmail.com

**Abstract.** There are over two billion mobile phone users around the world. There are also a large group of people with visual impairment. Can these people use mobile phones in the same way as other people? This paper tries to find out what problems and opportunities visual impaired people have with mainstream mobile phones today and their interface. A short introduction to mobile phone interfaces and visual impairments are presented. Following this, accessibility studies and interviews with users with visual impairment are presented. The conclusion shows that the interface itself is not always the issue, but also the hardware design and technical aids. The products on the market today can satisfy the needs of most of the people with visual impairment; however, the users want some improvements.

## 1 Introduction

*“Accessibility is a general term used to describe the degree to which a system is usable by as many people as possible.”* [1]

People have always had a need to communicate with each other. With telephones and PCs people want to make their lives more efficient and be available for friends, family and other important people. Today we use our mobile phones to do more than just calling each other. We can, for example, use them as cameras, pay with them and read the morning newspaper. When we moved into this mobile era we created a need that did not exist before, the need for faster communication. Today this need has been taken care of through more complex mobile phones [2].

These new complex mobile phones give the users new possibilities, but there are problems too. Mobile phones are small devices with a relatively small screen showing all the information. As people age, and with increasing age their vision, hearing, and touch are decreasing. There are of course younger people with these kinds of impairments also, but perhaps they are more welcoming to new technology. What is clear though is that an impaired person’s need to communicate is not less than any other’s need.

Technology is constantly evolving and new helping aids for people with special need have been invented. The ideas behind these aids are that a person with a handicap could feel less handicapped when using one of this [3]. Today it is hard

to find a mobile phone with all the accessibility features that is needed by an impaired person [4].

The main purpose of this paper is to explore the accessibility in mobile phones and its interface; meaning the screen and what is shown there. The question is if the mobile phones on the market today satisfy the need of a visually impaired person? Are there internal and external aids that can improve the usability of the mobile phone?

This paper will in section 2 and 3 introduce the reader to mobile phone interfaces and visual impairments. Section 4 discusses accessibility, technical aids and software. Section 5 includes two interviews with two persons having a visual impairment and this is also the evaluation section of the paper. In section 6 there will be a combined conclusion and discussion.

## 2 Mobile phone interfaces

When the mobile phone came 50 years ago [5], we moved into a new era and now we can communicate almost anywhere we would like. The development of the mobile phones today is proceeding faster than ever. The mobile phone companies strive for smaller phones, but at the same time more functionality. Interfaces in the form of a screen are very common in products today, and mobile phones are no exception. The interface of a mobile phone is primarily there to present information for the user. However, the interface could also be sounds, vibrations and other effects for helping the user to understand what happened or will happen. The figure below is a table that shows how the Nokia phones have changed over time (Fig. 1). We can see how the display shows more and more information, the software features are increasing while the volume and weight is decreasing.

Model	Nokia 1011	Nokia 2110	Nokia 6110	Nokia 6210	Nokia 6610
Year Introduced	1992	1994	1997	2000	2002
Display type	2x8 chars	3x10 chars + 2x6chars	84x48 pixels	96x60 pixels	128x128 pixels
Number of software features	406	378	1719	2777	3085
Volume	340 cm <sup>3</sup>	170 cm <sup>3</sup>	130 cm <sup>3</sup>	95 cm <sup>3</sup>	71 cm <sup>3</sup>
Weight	475 g	240 g	140 g	114 g	84 g

**Fig. 1.** This table shows how Nokia mobile phones changed over time. The table is a redesign of [6].

## 3 Visual impairments

People with visual impairments can be found all over the world. When talking about visual impairments we must understand that there are many different kinds, and people with these of course have different need [3].



In this paper, I will address three different populations of mobile phone users; people with low vision, people with color vision defects and blind people. These populations are described more in the sections below. I have chosen these populations because they are very different and I think there is a major difference in the optimal mobile phone interface for these populations.

### 3.1 People with low vision

This population includes people who are short-sighted, long-sighted, and people with astigmatism. These types of visual impairments can be, in most cases, corrected by glasses or contact lenses. If glasses or contact lenses cannot be used, surgery can be an alternative. Still, people sometimes forget their glasses or contact lenses and in those cases problems can occur.

Hallengren and Hed [3] state a couple of different design issues which is the importance of an interface that is easy to see or read, and with a good contrast in pictures and display. Furthermore, the text presented on the screen should be big and the font type is important. A sans-serif font like Arial is preferred because they are faster to read. The mobile phone should for example use sound feedback, to make the information from and to the phone even clearer. These kinds of design issues can be hard to measure. However, the designer could get important information from users and change the design before it is too late.

### 3.2 People with color vision defects

People with color vision defects are described as those having difficulties in distinguishing different colors [7, 8]. Color blindness is another word, but usually misused because there is only a very small percentage of people who cannot see any color at all. There are many different kinds of color vision defects. One kind could be which colors are hard to see, and another how many different colors that is difficult to distinguish. There is no treatment against color vision defects but there are some aids that can be used. Special types of aids are tinted filters that can be attached to glasses or used as contact lenses. This will not give the person normal color vision though, but could at least improve it.

### 3.3 Blind people

There are over 50 definitions of blindness worldwide [9]. The World Health Organization definition of blindness is when the better of the two eyes is less than 3/60. This means that the person cannot read the top letter of a Snellen chart [10] from a distance of three meters. Snellen charts are white boards with letters in different sizes and are used to measure visual acuity.

An interface designed for blind people could help them, even though a blind person cannot see the display [3]. The information from sounds and touch stimuli is important for the user. This paper does not examine how the hardware interface works, but general design issues could be the size of the buttons and

what shape they have. This is important for the user because it could help the user to distinguish the different buttons. Use of voice control could probably be another useful tool for this group of users.

## 4 Accessibility in mobile phone interfaces

The mobile phones today with smaller screens and more functionality can be a problem for people with different kinds of impairments [3]. A person with visual impairments may find it difficult to navigate an interface with all this technology and functionality. Still, we cannot ignore these people. They have a need to communicate just like the rest of the people. If we want to make it easier for these people, we must make the mobile phones more accessible.

As said before there are many people in the world using mobile phones daily. People use their phones to do many different things like calling other people, sending messages and connecting to the Internet. If a mobile phone lacks in accessibility that could limit the user to reach information available for example on the Internet.

According to Tomioka, and his report “*Universal Design Practices: Development of Accessible Cellular Phones*” [11] we have since 1997 used a more human-centered design process to design more user-friendly mobile phones. Furthermore, he states that mobile phones are very important for people with visual impairments just like the PC is. The mobile phone can give people with impairment a greater quality of life (QoL) if they can feel that the mobile phone is bringing something positive into their lives.

The aspect of QoL is something Nguyen et al. [4] are discussing in the article “*Accessible Mobile Phones*”. They think that a more accessible mobile phone for people with disabilities, will significantly improve their QoL. This should be reached through an increased range of accessible activities, but also things like independence, self-esteem and the feeling of being safe and secure.

### 4.1 Technical aids and programs

The larger mobile companies in the world hopefully try to improve the accessibility in mobile phones. However, smaller companies and researchers developing software and hardware for a better accessibility in mobile phones also [12, 13]. This section shows a couple of different technical aids and program that can enhance the mobile phone use.

Wagner et al. [12] investigated two different techniques that could improve the usability for a user with low vision. A mainstream mobile phone was used and no functionality was removed. The software enlarged the text on the button that was pressed, and showed it on the screen. The difference in techniques was that one technique added the number pressed to the dialing queue, and the other did not. The result showed that both these techniques improved the dialing accuracy. The limits with this study were that the sample group was small and that only one kind of mobile phone was used.

Abascal and Civit [13] has taken a more technical approach. They claims that improving the usability for visually impaired people and elderly people, is a question of finding the problems that can occur. They have stated five different scenarios when the mobile phone could help the user. These scenarios include problems and that is what has to be solved according to Abascal and Civit. They suppose technical solutions, and one example is GPS technology. The user could be both receiver and sender, and could in that case both find help and be found by someone.

In the Swedish magazine *“Allt om hjälpmedel”* [14] there is an article about Linda. Linda is deaf blind since she was a kid and uses a Braille display to her regular mobile phone. It is a relative small device but nothing you carry in your pocket. With this she can send text messages (SMS) typed with the device and the receiver who gets the message, will read it on a regular display. She can also read incoming messages and use other functions on her mobile phone with help from her Braille display. A big benefit with the system is that she can disconnect the Braille display from the phone. This could be useful when she has her interpreter with her, who can read the regular display. Another big benefit is that she is freer now, when she can communicate in a faster and more spontaneous way.

SonyEricsson and Nokia are two of the biggest mobile companies in the world. They both have special needs centers [15, 16] where users can buy mobile phones. The phones presented on these web sites are more or less regular mobile phones, but they will fit the visually impaired good. There are also different accessories available that will fit with a mobile phone from the same company.

## 5 Evaluating of the accessibility in mobile phones and their interfaces

The evaluation is in form of two interviews. The first one with a woman that has a significant visual impairment. She uses several different technical aids and has a long experience of mobile phones. She will in this evaluation represent the group with blind people, because she cannot see the display. The second interview is with a man who is long-sighted and uses glasses. He uses no technical aids, but will describe the problems he has with his mobile phone. He will represent the group with low vision.

The interview is a semi-structured interview, which is a relative open interview focused on a two-way communication. I chose this kind because I had questions I really wanted answers on, and because I wanted a communication with the user. With this kind of interview it is also easier to come up with new questions during the interview. The questions below are some of the ones I had for Kristina, the first user. (Author’s translation from Swedish)

- What is your age, how long time has you been visually impaired?
- What kind of aids do you use today?
- Do you have a mobile phone, and for how long have you had it?

- Do you have a mobile phone, and for how long have you had it?
- What do you think is most important; the software or the hardware?
- What kind of improvements in the modern mobile phones do you want to see?

Some of these questions were used for Kjell, the second user, as well. Why I did not use all of them was because they did not fit in the interview, and therefore I had to construct new questions during the interview with him.

### 5.1 Kristina Strindlund

Kristina is 59 years old and has a significant visual impairment (see Figure 3). She has been visually impaired since 1990, so she has experience from seeing and consequently knows what things look like. She works as a librarian at Umeå University and uses different technical products every day. In her office there are many different helping aids. The first thing noticed notice is a CCTV, which is like a magnifier for different things. This machine presents the magnified information on a screen and can also invert the colors, if you for example prefer white text on black background. She is listening to many audio books every month, and for this she has two different machines. One of them is a little bit bigger and the other is more like a small MP3-player. These machines read Daisy records, which is like an audio book but more compressed data. To her computer she uses a screen reader named JAWS. This is software that reads the screen for the user and can also magnify the screen up to 32 times. Kristina also uses a mobile phone. She got her first phone 1992 and has since that year used many different kinds. She cannot see the screen but has nevertheless used any external aids to help her. She says that the interface is not important for her, because she cannot see the screen. However, it could be good when new software need to be installed or configured. Right now she is using a Nokia 6630 which she has owned for two years (see Figure 2).

In the beginning she thought that the rounded keys could be a problem, but now she is using them with no problem. With her Nokia she uses software named Mobile Speak. This software is a screen reader which reads the screen and thereafter speaks the text to the user. She demonstrates the navigation in the interface and the address book. The voice that comes out is high and clear and there is absolutely no problem to understand it. She has not used any magnifiers for mobile phones, because she cannot see the screen anyway. She says though that her friends who can see more than her have used it and are pleased with it.

Kristina thinks that the mobile phones today are more accessible and useful than her first mobile phones in the early 90's. Several different authors [4, 11] on the topic state that a mobile phone will improve the QoL. Kristina can just agree with this and says that especially youths can feel more connected and free when they can send SMS. She is not using all the functions available in the phone. Making and receiving calls, using the address book, reading SMS and checking the call list is the functions she is using. When I asked her if she ever tried to connect to the Internet through her mobile phone, she is just smiling. She says



**Fig. 2.** Kristina's telephone, a Nokia 6630. As the picture shows, no external aids are used.

it is possible but easier using her computer. But in the other hand, she does not always have her computer with her.

Even though this mobile phone is working well for Kristina, there could be some improvements. She thinks that there are too many clicks before the desired function is reached. She has a friend that has shortcuts to more or less all the functions. This is something that she wants to combine with an even more usable interface. Once she tried a Nokia external keyboard, but she thought it was too big and difficult to use. Another problem she has is that the mobile speak does not work with and hands-free. She feels that this is a kind of integrity problem. When she is calling someone, people around her can hear what number she is calling. And when the software is reading a SMS for her, the same situation can occur. After all, she is pleased with her mobile phone and that is the most important.

## 5.2 Kjell Andersson

Kjell is 54 years old and have used glasses for five years. He is long-sighted and has problems when reading and seeing things in a close range. He considers himself used to mobile phones and has used them for around 15 years. During this interview he is not wearing his glasses. The mobile phone is a SonyEricsson K750i with no added technical aids (see Figure 4), and it is not his own mobile phone. This because I wanted him to see something that he is not used to.



**Fig. 3.** Kristina Strindlund.

When he is looking at the mobile phones from normal reading distance, he can only see the numbers on the buttons, not the letters. They are blurred and he has to move the mobile phones closer to distinguish the letters. The sub menus are also blurred, but he can read them if he concentrate his eyes (see Fig. 4 right picture).

The icons are showed when you move into the first menu (see Figure 4 left picture). These are no problem for Kjell to see and he can without any bigger problems recognize what they mean. He thinks they have a good design and contrast. The text above them is harder to see. Like the other texts, it is blurred, but with some concentration he can read it.

Like Kristina said in her interview, Kjell also wants to see some improvements. Kjell drives his car a lot in his job. And when doing that he does not carry his glasses. Sometimes when he needs to make a call he cannot see the display correctly, and therefore he has called the wrong number a couple of times. He thinks this could be corrected by bigger text or maybe a better working voice dialing function than he has. Bigger text is also something suggested by Hallengren and Hed [3] for people with low vision Even though he could see the icons in the interface, he wants them bigger or lesser.



Fig. 4. Left: The icons in the SonyEricsson K750i. Right: The text in the sub menus.

## 6 Conclusion

According to Fruchterman [17] the majority of people with disabilities do not use the aids available. People often reject these aids because they are expensive, difficult to use, not enough information about them or just that the user do not want to use special aids.

I think though, that the young population today that in any way are impaired will have a greater need of these kinds of products. The mobile phone today is an important tool for youths, which is certified by both Kristina Strindlund and Linda [14]. They can feel more free and maybe feel a little bit less impaired.

In the section with different visual impairments I had a population of people having color vision defects. I did not have any test person for this group because of a couple of different reasons. First of all I could not find a person with this impairment, which I really wish I had. I thought I maybe should try to make a test for myself but according to Hoffman [18] that is not good. He says that doing an interface test related to users with color vision defects, without using real users will not give the right accuracy.

Statistics shows that 1 to 8 percent (depending on race) of the males in the world have some kind of color vision defect, when only about 0.4 percent of the females have it [7]. This means that there is a relative big part of the males that could have problems with an interface not designed for these people. This should of course be considered when designing an interface for a mobile phone.

Without having any user for testing this time, I will present a couple of thoughts. When designing interfaces for people with color vision defects, Halengren and Hed [3] says that these users are not in need of a simplified mobile phone or interface. However, they state that the interface should not have a

color as single carrier for information. This is something Hoffman also states [18]. He suggests that if the designer wishes to make the interface equal to all users, red/green and blue/yellow color coding should be avoided. Both Hoffman and Hallengren and Hed states that if color is used for presenting information, they should have a distinct contrast to the background and other colors used.

Statistics shows that 1 to 8 percent (depending on race) of the males in the world have some kind of color vision defect, when only about 0.4 percent of the female have it [7]. This means that there is a relative big part of the males, that could have problems with an interface not designed for these people. This should of course be considered

Nguyen et al. [4] says that some Internet-based applications, such as sending and receiving e-mails information searches are all visually based and therefore exclude blind consumers. I think that is something that could be discussed. According to the interview with Kristina, all functions are available with her software "*Mobile Speak*". I think that she could use those function, if she learns how to use them.

Hallengren and Hed [3] states that low vision users will feel no or a little enhancement with an improved interface. However, small changes like clearer interface, bigger text and better designed icons could make the interface more usable. These changes are something Ornella and Stéphanie [19] also agree with in their paper "*Universal Design for Mobile Phones: A Case Study*". Kjell is asking for this in the interview, and this looks like something that could be useful when designing for visually impaired people.

Even though both Kristina and Kjell are pleased with their mobile phones there could of course be improvements. Fruchterman [17] have a solution to why people reject the technology available. He says that if we could build more adaptive functions into standard devices that are used by many people we could overcome these problems. If the market could do this the products will become cheaper, look more like standard devices and will be easier to use. How far we are from this is not yet clear.

Maybe we can try to formulate an answer to the questions from the introduction now. Can the mobile phones on the market today satisfy the need for a visually impaired person? I would say yes. Maybe not in its original state but with more or less modification. Which leads us to the second question, whether there are internal and external aids that can improve the usability of the mobile phone? I would like to say yes again, according to Kristina, who are using a couple of different aids we can see that her software makes it possible for her to use a mobile phone. However, they can of course be improved.

This paper began with the aim on mobile phone interfaces. Now I would like to say that the interface itself may not always be the issue. I think that the whole mobile phone kit with the physical phone, software and the external and internal aids are the issue. This issue is not yet solved completely, but we seem to be on the right way.



## References

1. Wikipedia: (Accessibility) <http://en.wikipedia.org/wiki/Accessibility>, accessed 2007-04-27.
2. Rydberg, M., Winbo, P.: Mobiltelefonens historia. Department of Computer Technology, Mälardalens University (2004)
3. Hallengren, E., Hed, I.: A simplified user interface for mobile phones. Master's thesis, Lund University, Sweden (2004)
4. Toan Nguyen, A.D., Garrett, R.: Accessible mobile phones. In: ARATA Conference. (2001)
5. Wikipedia: (History of mobile phones) [http://en.wikipedia.org/wiki/History\\_of\\_mobile\\_phones](http://en.wikipedia.org/wiki/History_of_mobile_phones), accessed 2007-04-20.
6. Christian Lindholm, T.K., Kiljander, H.: How Nokia Changed the Face of the Mobile Phone. McGraw-Hill, New York, NY, USA (2003)
7. : (University of illinois eye center, eye facts) <http://www.uic.edu/com/eye/LearningAboutVision/EyeFacts/>, accessed 2007-04-26.
8. Wikipedia: (Color blindness) <http://en.wikipedia.org/wiki/Color-blindness>, accessed 2007-04-26.
9. : (Cochrane eyes and vision group) <http://www.cochraneeyes.org/glossary.htm#B>, accessed 2007-04-19.
10. Wikipedia: (Snellen chart) [http://en.wikipedia.org/wiki/Snellen\\_chart](http://en.wikipedia.org/wiki/Snellen_chart), accessed 2007-04-19.
11. Tomioka, K.: Universal design practices: Development of accessible cellular phones. In: Designing for the 21st Century III - An international conference on universal design, Rio de Janeiro, Brazil (2004)
12. Jennifer Wagner, G.C.V., Sesto, M.E.: Improving the usability of a mainstream cell phone for individuals with low vision. *Journal of Visual Impairment and Blindness* (2006) 687–692
13. Abascal, J., Civit, A.: Universal access to mobile telephony as a way to enhance the autonomy of elderly people. In: WUAUC'01: Proceedings of the 2001 EC/NSF workshop on Universal accessibility of ubiquitous computing, New York, NY, USA, ACM Press (2001) 93–99
14. Udd, L.: Nu kan jag läsa mina sms. *Allt om hjälpmedel* (5) (2005) 12–15
15. SonyEricsson: (Special needs center) <http://www.sonyericsson-snc.com>, accessed 2007-02-27.
16. Nokia: (Nokia accessibility) <http://www.nokiaaccessibility.com/>, accessed 2007-02-27.
17. Fruchterman, J.R.: In the palm of your hand: A vision of the future technology for people with visual impairments. *Journal of Visual Impairment and Blindness* (2003) 585–591
18. Hoffman, P.: Accommodating color blindness. *Usability Interface* (2) (1999)
19. Ornella, P., Stéphanie, B.: Universal design for mobile phones: A case study. In: CHI 2006 - Work-in-Progress, Montréal, Québec, Canada (2006)



# Evaluation of Quality of Service Performance in Wireless Local Area Networks

Muhammad Shahid Manzoor

Department of Computing Science  
Umeå University, Sweden  
`int05smr@cs.umu.se`

**Abstract.** The IEEE 802.11 standard is currently the most successful WLAN technology in the world, due to its cheap cost and easy installation. But the 802.11 lacks the support of QoS. The QoS refers to the quality of the data traffic over a network. Modern multimedia applications require strict QoS requirements in terms of bandwidth and delay. Unfortunately due to lack of QoS support, 802.11 is unable to fulfil these requirements. A recently released version of 802.11, called 802.11e, introduces QoS support by differentiating applications based on their QoS requirement. This paper presents an overview of 802.11 and 802.11e. It discusses limitations of 802.11 in providing QoS support and how these limitations are overcome in 802.11e.

## 1 Introduction

The IEEE (Institute of Electrical and Electronics Engineers) introduced the 802.11 wireless local area network (WLAN) standard in 1997 [1]. Since then, it has become an immensely popular wireless technology. The 802.11 standard defines MAC (Medium Access Control) layer and physical layer (PHY) for a WLAN. It supports a maximum transmission data rate of 2 Mbps. The IEEE made improvements to the 802.11 WLAN and released two new versions IEEE 802.11a [2] and IEEE 802.11b [3]. These new versions support a data transmission rate of 11 to 54 Mbps in the 2.4 GHz frequency band. IEEE 802.11 WLAN has attained enormous popularity due to its low cost and easy installation, but unfortunately it does not support the QoS. The QoS is a networking term which defines a set of attributes such as bandwidth usage, packet loss, delay and throughput. Over a network this set of attributes shows the quality of data traffic. QoS requirements of the data traffic differ from application to application. A network that fulfils the requirements of applications is called a QoS supported network. QoS is evaluated by available bandwidth, delay in data transfer and data loss. The QoS is categorized in three terms bandwidth, data Loss and delay. Basically the MAC protocol of 802.11 works on first come first serve basis, so all types of applications are served the same way ignoring the QoS requirements of the data traffic. Modern multimedia applications are very sensitive to the availability of bandwidth, and delay in the data transfer. Some

examples of such sensitive applications are, video and audio streaming, internet telephone and online network games. Due to the incapability in providing QoS, a lot of research work has been done in recent years to add QoS support to IEEE 802.11 networks. At present, IEEE is working on 802.11e which is designed to provide QoS support. This paper discusses 802.11, its weaknesses in providing QoS support and how QoS is introduced in 802.11e. The remainder of this paper is organized as follows. Section 2 presents an overview of 802.11 and its medium access mechanism. Section 3 discusses QoS and QoS limitations of 802.11. Section 4 presents an overview of 802.11e, its access mechanism and how QoS is supported by introducing service differentiation. Finally, summary and conclusion of the paper is discussed in section 5.

## 2 An Overview of IEEE 802.11

The IEEE 802.11 WLAN standard was released in 1997 [1]. It explains specifications for MAC and physical layer. There are three different types of physical layer specifications explained, which are FHSS (Frequency Hopping Spread Spectrum), DSSS (Direct sequence spread spectrum) and IR (Infrared). FHSS and DSSS physical layers work in the license free 2.4GHz Industrial, Scientific and Medical (ISM) frequency band. These three layers provide transmission data rate of 2 Mbps. After two years, the IEEE introduced two new versions IEEE 802.11a and 802.11b. The IEEE 802.11a is based on Orthogonal Frequency Division Multiplexing (OFDM) and it supports data transmission rate from 11 to 54 Mbps and works in 5GHz frequency band. The IEEE 802.11b operates in 2.4 GHz frequency band and it is also based on DSSS. It also supports 11 to 54 Mbps data transmission rate. In 2003, the IEEE released 802.11g [4] by improving physical layer descriptions of IEEE 802.11b in 2.4 GHz frequency band. The IEEE 802.11g standard supports data transmission rate up to 54 Mbps [5]. Nowadays, the IEEE 802.11 is one of most popular wireless technology of the world and is installed in offices, hotels and airports. The reason in gaining great success is low cost and easy installation. The IEEE 802.11 standard consists of two different basic structures. These structures are Basic Service Set (BSS) and Independent basic Service Set (IBSS) [6]. The BSS contains a number of wireless stations connected with Access Points (AP). The AP is responsible for communication between wireless stations. In IBSS, wireless stations can communicate within given transmission range provided with each other. The basic access method is called Distributed Coordination Function (DCF) and is described in the next section.

### 2.1 Distributed Coordination Function (DCF)

The DCF is the basic access method of 802.11 WLAN standard. It is based on Carrier Sense Multiple Access (CSMA), which works as listen before talk scheme. If a station wants to transmit a frame, it senses the medium. A station starts transmission of frames when the medium is sensed idle for DCF Inter

Frame Space (DIFS) time period. The ACK (Acknowledgement) frame is sent back for acknowledgement, when a receiver station receives a frame after Short Inter frame Space (SIFS) time period. The IEEE specified three IFS (Inter frame Space) time periods which are Short Inter Frame Space (SIFS), Distributed Inter Frame (DIFS) and Point Inter Frame Space (PIFS) to control the medium access. The DIFS is the largest Inter Frame Space and SIFS is the shortest Inter Frame Space. Depending on the priority of the frame exchange sequence, consecutive frame transmissions can be differentiate by these inter frame spaces, greater the priority of frame exchange sequence shorter the inter frame space used between frames. To determine whether a medium is busy or idle there are two kinds of carrier sensing which are Physical Carrier Sensing and Virtual carrier sensing. When wireless channel senses itself at physical layer it is called physical carrier sensing. Virtual Carrier Sensing is used at MAC layer. When a station receives a frame which is not addressed to itself, it investigates the time from frame header. Frame header defines explicitly the time require for the transmission of frame. Then it postpones medium access for that time period [7, 8]. Figure 1 illustrates DCF basic mechanism.

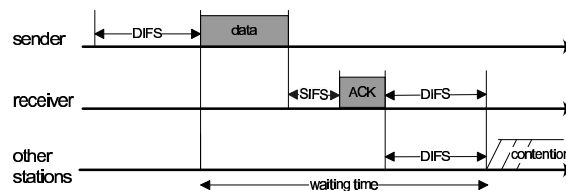


Fig. 1. DCF basic access mechanism (Courtesy of Jahanzeb Farooq [7]).

Collisions occur when two or more station finds the medium idle and transmit frame at same time. To prevent this situation stations have to wait and choose random back off time. Back off time indicates the time duration for which a station has to wait before starting transmission after waiting for DIFS period.

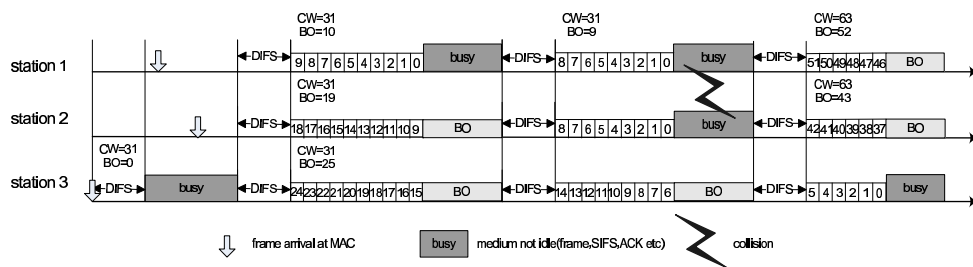


Fig. 2. DCF access mechanism with backoff procedure (Courtesy of Jahanzeb Farooq [7]).

When medium becomes idle, station start decreasing its backoff time in DIFS time period. As backoff time decreased to zero, station starts frame transmission. If the medium is found busy, station stops the backoff time and resumes it after medium becomes idle again for the DIFS period. The random backoff time is used to avoid collisions. This method is called Collision Avoidance (CA) and hence the whole mechanism is called CSMA/CA. The random backoff value is uniformly drawn from range  $(0, CW)$  where  $CW$  is called Contention Window. The contention Window ( $CW$ ) is set to minimum  $CW_{min}$ . The  $CW$  size becomes doubled every time a transmission is unsuccessful, until it reaches to its maximum size  $CW_{max}$ . The  $CW$  is reset to  $CW_{min}$  after every successful transmission. Figure 2 illustrates the access mechanism of DCF.

QoS and limitations of IEEE 802.11 standard are discussed in next section.

### 3 Quality of Service and Limitations of IEEE 802.11

The QoS is a networking term which defines a set of attributes i.e. bandwidth use, jitter, delay, packet loss, and throughput. This set of attributes defines the data traffic quality over of network. The QoS requirements differ from application to application. An application requires certain QoS requirements and a network which meets with each application's QoS requirements is called QoS supported network. The QoS requirements can be categorized in three types such as bandwidth, delay and data loss [9, 10, 7].

1. **Bandwidth:** Bandwidth is the most significant factor that specifies data transfer quality in given time period. Application can transfer data in large amount if it receives higher bandwidth. Bandwidth sensitive applications need consistent data transfer rate and can be affected by any reduction in bandwidth. Hence result comes in the form of unwanted delay and data loss. Multimedia application like internet telephony and video conferencing are called bandwidth sensitive applications [9, 10, 7].
2. **Data Loss:** Data oriented applications such as email, web pages are usually loss sensitive and they tolerate low bandwidth and infrequent delays but demand reliable data transfer. Multimedia applications are categorized as bandwidth and delay sensitive but usually are loss-tolerant and demand bandwidth and delay assurance. Data loss in these applications results in distortion of data [9, 10, 7].
3. **Delay:** Multimedia applications such as internet telephony, video conferencing and network games are very delay sensitive. Any increase in delay severely damages their performance [9, 10, 7].

The major problem with the IEEE 802.11 standard is that, it is based on best effort service model. It serves all kind of applications on first come and first serve basis. It can not differentiate application on the basis of the QoS requirements. The bandwidth sensitive applications are given no priority over delay sensitive applications in terms of bandwidth. Similarly, delay sensitive applications are not given any priority over bandwidth sensitive applications in terms of delay. Due to

incapability in providing QoS support to applications which have different QoS requirements, the IEEE 802.11 working group is currently working on the IEEE 802.11e standard which is called QoS supported network. Next section presents an overview of the IEEE 802.11e.

## 4 An Overview of IEEE 802.11e

Presently, the IEEE is working on an improved version of IEEE 802.11 MAC called the IEEE 802.11e [11]. It consists of priority mechanism in order to support the QoS. It serves data traffic to all kind of applications according to their QoS requirements. Since different applications have different QoS requirements, therefore applications are classified in four Access Categories (AC). Every frame with a certain priority of data traffic is assigned to an access category. AP (Access Point) and stations that offers the QoS services are called QoS Access Point (QAP) and QoS station (QSTA) and the Basic Service Set is called QoS Basic Service Set (QBSS). Next section explains EDCA (Enhanced Distributed Channel Access) which is basic mechanism in 802.11e to support QoS.

### 4.1 Enhanced Distributed Channel Access (EDCA)

The EDCA mechanism is an improved version of the DCF mechanism that facilitates distributed differentiated medium access to wireless channels by utilizing different priorities and with the help of multiple access categories (ACs). The EDCA defines four Access Categories (ACs) and is designed to handle eight different traffic priorities for several types of data traffic. The access opportunity differentiation is given by Arbitration Inter-Frame Space (AIFS) regardless of the constant DIFS, and different values for the minimum/ maximum contention windows for the backoff extractions. Furthermore, for each AC a different set of parameters is utilized to access the medium. These parameters are known as EDCA parameters. The four access categories (ACs) are AC\_BK, AC\_BE, AC\_VI, and AC\_VO

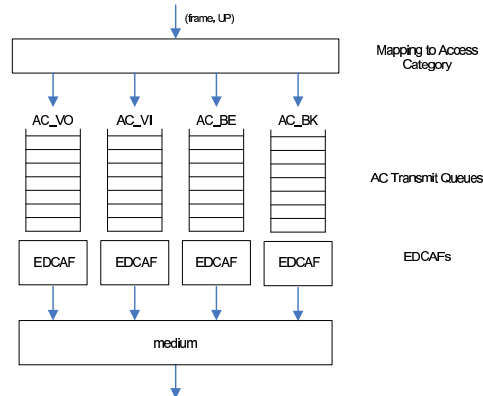
1. AC\_BK for background traffic
2. AC\_BE for best effort traffic
3. AC\_VI for video traffic
4. AC\_VO for voice traffic

AC\_VO is the highest priority and AC\_BK is the lowest priority.

When a frame arrives at MAC layer, it contains a priority value which is called User Priority (UP). UP of the frame is then mapped to corresponding AC. Table 1 shows UP to AC mapping.

### 4.2 Enhanced Distributed Channel Access Function (EDCAF)

EDCAF is the improved version of DCF and contends for the medium access on similar rules of CSMA/CA. There are four transmit queues, one for each EDCAF. Each of the four EDCAF works individually.



**Fig. 3.** Four ACs, each with its own queue, AIFS, CW and backoff timer (Courtesy of Jahanzeb Farooq [7]).

There are two EDCA parameters which are utilized for contending access of the the medium [7]. These parameters are:

1. Arbitration Inter Frame Space (AIFS) AIFS is the time period during which medium is found idle before transmission. Higher priority ACs have smaller AIFS values and lower priority ACs have greater values. Higher priority ACs have to wait for less time before starting transmissions.
2. CWmin and CWmax: They vary from AC to AC. Lower priority AC have greater CWmin and CWmax values compared to higher ACs. Therefore, higher priority ACs most of the time get smaller backoff time and therefore have to wait for less time [9, 7, 12].

AC	CWmin	CWmax	AIFSN
AC_VO	7	15	2
AC_VI	15	31	2
AC_BE	31	1023	3
AC_BK	31	1023	7

**Table 1.** Default EDCA parameter values (Courtesy of Jahanzeb Farooq [7]).

Figure 5 illustrates the EDCA access mechanism. This figure explains that EDCA access mechanism has different sets of parameters for different ACs. As soon as for AIFS time period, when medium becomes idle, EDCA picks a random backoff value and begins reducing its backoff timer. When backoff timer arrives at zero, transmission is started. AC which has greater priority obtains more amount of bandwidth by transmitting more frames as compared to the AC which has less priority. For greater priority ACs like AC\_VI and AC\_VO, wait only for a



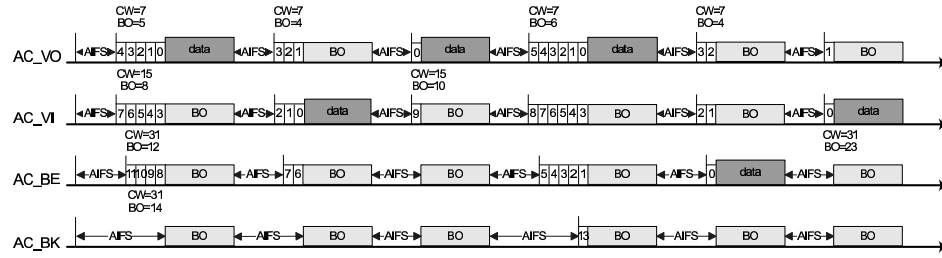


Fig. 4. EDCA access mechanism (Courtesy of Jahanzeb Farooq [7]).

short AIFS time period and start reducing their backoff timers. But in the case of less priority ACs like AC\_VO, it has to wait for a long AIFS time period. One bigger factor is that greater priority ACs has small maximum and minimum contention window (CW) sizes as compared to less priority ACs which have large size contention windows, so they have to wait more AIFS time period for the transmission of frames [7, 10, 9].

## 5 Evaluation

The Global Information System Simulator (GloMoSim) is used as scalable environment for mobile and wireless networks [13]. It is written in PARSEC which is a C based language. Simulations are used to compare the IEEE 80211 DCF and 80211e EDCA in the context of QoS schemes.

### 5.1 DCF vs. EDCA Performance Comparison

A simple scenario is simulated in order to provide with a comparison of 802.11 DCF and 802.11e EDCA. In this scenario stations are used from 1 to 10 which are transmitting streams related to all four types of traffic, as explained in Table 1. The objective is to examine the performance of individual ACs, i.e., how traffic related to a specific AC is served and also to evaluate the performance of the IEEE 802.11e EDCA and 802.11 DCF. In 802.11e traffic streams are served related to their ACs but in the IEEE 802.11 DCF all traffic streams are served with same priority.

Figure 5 shows the throughput results for the 802.11 DCF which explains throughput results for four different types of traffic streams produced by a station. These four traffic types are represented as video, voice, best effort and background, on the basis of their characteristics, i.e. bit rate, packet size and interval. As it has been discussed earlier, that the IEEE 802.11 is based on best-effort service model. It sends frames from sender to receiver as soon as possible. Since there is no support for service differentiation, it serves all four types of data traffic similarly apart from their QoS requirements. As shown in figure, regardless of the QoS requirements of different traffic streams, all four traffic streams are served in the same way and therefore experience same amounts of degradations

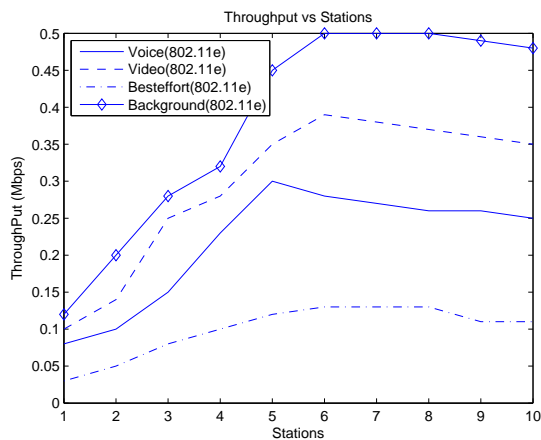


Fig. 5. Simulation results for IEEE 802.11.

in bandwidth. This clearly shows the basic problem with 802.11, i.e. all types of traffic are treated equally; there is no means of prioritization/differentiation. It is clear from the results that all types of data traffic get equal amount of bandwidth and suffer from variations in bandwidth and same amount of delays as the network becomes congested.

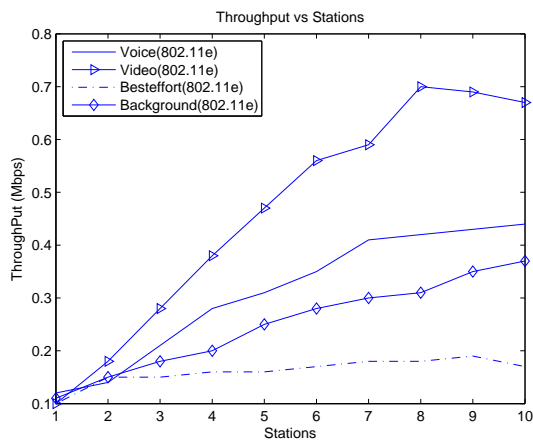


Fig. 6. Simulation results for IEEE 802.11e.

Figure 6 shows throughput results for 802.11e EDCA. It is clear from the results that since 802.11 DCF serves all types of traffic streams in the same way, IEEE 802.11e EDCA effectively provides service differentiation through different

ACs. Further 802.11e allows the high priority traffic voice and video to receive higher and stable throughput through its service differentiation mechanism. As compared to 802.11 DCF, the IEEE 802.11e EDCA provides constantly better throughput to both video and voice data traffic streams. We can observe that both voice and video traffic streams receive surprisingly better throughput as compared to 802.11. It is also reconfirmed that 802.11e is better by observing the severe drop of background and best-effort data traffic streams. In 802.11e, Higher priority traffic (voice and video) have to suffer from small delays in comparison with lower priority traffic (best-effort and background).

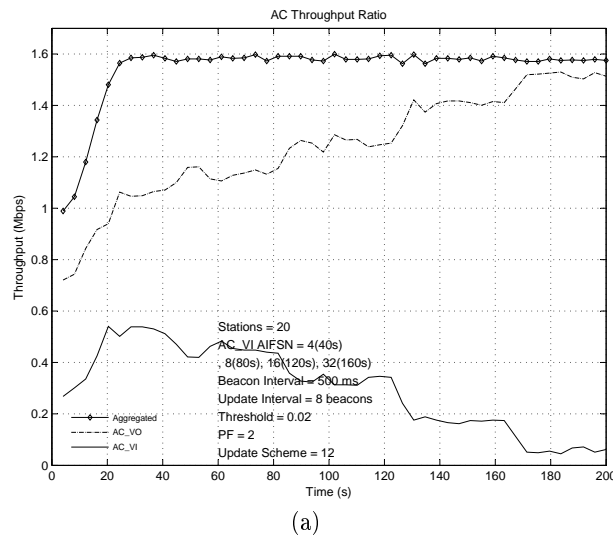


Fig. 7. Achieving desired service differentiation with AIFSN.

Figure 7 shows results for another scenario. It shows how the desired differentiation can be achieved by adapting EDCA parameter values ( $CW_{min}$ ,  $CW_{max}$ , AIFSN) dynamically. There are 20 stations, each station transmitting two streams, one for AC\_VI and one for AC\_VO. The figure shows how the throughput of AC\_VO is improved in response of gradual increase in AC\_VI AIFSN. AC\_VI AIFSN value is initially set to 2, and then doubled every 40 seconds, ending at 32. It is seen that the AC\_VO throughput raises rapidly while that for AC\_VI is almost starved once set to 16 and 32. It further proves the effectiveness of 802.11e QoS scheme, i.e., how effectively the QoS parameters are used to achieve desired service differentiation on per stream basis.

## 6 Summary and Conclusion

The IEEE 802.11 standard is one of the most popular WLAN technology. Due to easy installation and low cost, it became most popular wireless technology of the world. The fundamental access method of 802.11 is called DCF. DCF is based on CSMA (Carrier Sense Multiple Access). Station senses the medium idle for DIFS time period before starting transmission and when medium becomes busy, it chooses random backoff value and waits for the medium until it becomes idle again. The random backoff value is taken in the range of  $(0, CW)$  where  $CW$  is called contention window. The main problem with DCF is that, it does not support for QoS. The QoS is a networking term which specifies a set of attributes like bandwidth, delay and data loss. QoS requirements are different for every application. Some applications are very sensitive to variation in bandwidth, others are sensitive to delay. Unfortunately DCF treats all type of applications in the same way. It does not differentiate applications on the basis of their QoS requirements. An upcoming version of the IEEE 802.11 is called the 802.11e, which provides support for QoS. Access mechanism of 802.11e, called EDCA, introduces access categories. Data traffic from different application are mapped to these access categories on the basis of their QoS requirements. EDCA uses different medium access parameters for different access categories to support the QoS. This paper also concludes the the performance of the EDCA and DCF mechanisms in order to support QoS for multiple type of applications. The simulation results show that all types of data traffic are served in a same way in 802.11 DCF. But in IEEE 802.11e every application is served on the basis of its QoS requirement and is mapped to corresponding AC. This paper also concludes how every application achieves its desired distributed differentiation according to its QoS requirement.

## References

1. : IEEE Std. 802.11, Part 11: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (1997)
2. : IEEE Std. 802.11a, Supplement to Part 11: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 5 GHz Band*. (1999)
3. : IEEE Std. 802.11b, Supplement to Part 11: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*. (1999)
4. : IEEE Std. 802.11g, Supplement to Part 11: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band*. (2003)
5. Lindgren, A., Almquist, A., Schelen, O.: Evaluation of Quality of Service schemes for IEEE 802.11 wireless LANs. Proceedings of the 26th Annual IEEE Conference on Local Computer Networks (LCN 2001) (2001) 348–351
6. Deng, D., Chang, R.: A priority scheme for IEEE 802.11 DCF access method. IEICE Trans. Commun **82** (1999) 96–102

7. Farooq, J., Rauf, B.: *Implementation and Evaluation of IEEE 802.11e Wireless LAN in GloMoSim*, Department of Computing Science, Umea University, Umea, Sweden (2006)
8. Bianchi, G.: *Performance Analysis of the IEEE 802.11 Distributed Coordination Function*. In: IEEE JSAC. Volume 18. (2000) 535–547
9. Banchs, A., Azcorra, A., Garcia, C., Cuevas, R.: *Applications and Challenges of the 802.11e EDCA Mechanism: An Experimental Study*. In: IEEE Network. Volume 19. (2005)
10. Nilsson, T.: *Resource Allocation and Service Differentiation in Wireless Local Area Networks*. (Licentiate Thesis, Dept. of Computing Science, Umeå University, June 2005)
11. : IEEE 802.11e/D13.0, Draft Supplement to Part 11: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements*. (2005)
12. Jong-Deok, K., Chong-Kwon, K.: *Performance Analysis and Evaluation of IEEE 802.11e EDCA*. *Wireless Communications and Mobile Computing* **4** (2004) 55–74
13. Bagrodia, R., Zeng, X.: *Glomosim, A Library for the Parallel Simulation of Large Wireless Networks*. Proceedings of the 12th Workshop on Parallel and Distributed Simulation (PADS'98) (1998) 154–161



# Autonomous Peers Collaboration

Davide Neri

Student in Computing Science  
Università degli studi di Modena e Reggio Emilia, Italy  
Umeå University, Sweden  
neridavide@gmail.com

**Abstract.** In the present paper will be delineated a new method that proposes a way to create a collaboration between several devices. This kind of collaboration will be based on the sharing and utilization of services. Devices are connected through a Bluetooth network. Every device is a peer with the others and can both use other devices' services, and share services. The method doesn't propose just a way to create a collaboration between devices, but also a method to make them collaborate completely autonomously. The project aims to be completely cross-platform and compatible with every kind of device. In order to achieve this compatibility, the collaboration is operated with XML structured messages.

## 1 Introduction

Imagine a scientific laboratory that works on several kinds of scientific tests; the lab has been asked, by a customer, to operate a new kind of test. The lab has never done this test before so a new testing apparatus might need to be bought in order to perform this kind of test. This is a new device, completely unconnected from the other devices in the laboratory, and, probably, incompatible with most of them. Therefore scientists have to operate tests with this new device, manually write down results, probably also to rewrite the results in a database on a computer, or use them in other devices that can operate complex calculus, or that needs these results to compute other parts of the test. There could be several possibilities and innovations by these scientists, but there will be a common problem of incompatibility and non-interoperability between devices. Now imagine what would happen if all these devices could communicate, help themselves, share services and resources with each other and do all this completely autonomously. This improves the efficiency, simplifies the level of complexity and reduce the operators' error. Computers have definitely changed modern society and the way people live [1]. Computer's ability to be able to perform electronically and autonomously simple or complex operations makes them able to handle large amount of data, and do calculations that are completely beyond the ability of the human brain. Despite the many functions of the computers, some situations still demand the attention of humans. Sometimes there are devices (or software's) which need to be installed or situations that might demand the use

of two or more devices which are dependent on each other. Results from one device might need to be transferred to an unconnected device manually, and this might be time consuming. There is a need for an online or multi-connected operating system where the results from one system are automatically transferred to another and thus improving efficiency and reducing the time. Several solutions exist today such as the use of ad-hoc built software, but this is expensive; or solutions such as connected devices from the same company can be used, but these could be hard to maintain and might restrict the user to the software or the company producing the devices. The present paper proposes a new method to solve this problem. Very little literature on related work could be found. The only project that shares similar aims is the JXTA project [2]. JXTA targets interoperability, platform independence, and ubiquity; its main objective is to allow devices to collaborate through a P2P network. JXTA has been created by Sun Microsystems and, despite from the present project, is completely based on Java. Only a new implementation, JXTA-C/C++, allows a non-Java device to utilize this technology.

The aim of this paper is to propose a theory or scheme to solve the problem of incompatibility and to produce devices which are autonomous and work together without human intervention or any kind of barrier due to different architectures, platforms, or languages. This will be called Autonomous Peers Collaboration (APC).

## 2 Collaboration

In order to understand how an Autonomous peers collaboration can be possible, we have to gradually understand the meaning of each of these three words. The main concept, that is essentially the base of the entire project's idea, is the collaboration. This paper proposes just about a special kind of collaboration. We can define that a collaboration between two entities exists when they work jointly on an activity or project [3], that is when they shares an own contribution for the development that project.

**Introduction to object-oriented software** In the early years of computer science software was created on a single long and complex block of instructions. The more difficult this computational complexity became, the more programmers from all over the world, realized solutions for this problem. One of these solutions is to split the source code in several smaller blocks [4]. Today computer science is oriented more and more toward the collaboration between files, also called "objects". Object oriented languages, such as Java, have improved the management of big projects: a Java project can be split in several objects; each one of them having a specific function which can be easily utilized from the other objects. The developer creates every object individually with all its own private and public functions. Every object has a own role, useful to the project, and is enabled to use the public functions of the other objects (and vice-versa). This is a collaboration: more entities (objects) sharing a contribution for the



development of a job. This is the model of collaboration that the APC project uses.

The model of collaboration of the object oriented software is limited, that for most of the times two objects to communicate they must be built on the same language and work on the same platform. This means that usually this kind of collaboration can exist only between objects that are on the same machine. This is too restrictive. Sun Microsystems has found a way to partially solve this problem in Java: the Java Virtual Machine (JVM). JVM is an abstract computing machine implemented as an application in computers. This application is a superstructure that enables Java software to work on every machine where the JVM is installed. It is a layer that works between the Java software and the operative system [5]. But this solution doesn't solve another part of the problem: Java objects are basically able to communicate only with other Java objects; they're not able to interface themselves with other kind of objects. Several solutions to this problem have been created, but most of them are not easy to use and are specific for only one kind of language; this is not very useful. A real solution to the problem has been introduced by Web services.

## 2.1 Short Introduction to Web Services

The web services technology enables the collaboration between objects present on different computers, even if they use different platforms, different operating systems, and even if they're protected by careful firewalls. This technology is not a new programming language, is not a new standard, is just a set of rules that defines the way hosts should collaborate. Web services are based on XML (eXtended Markup Language) and everything in a web service is developed in this language. The basic idea of a web service is to create a common interface, in XML, for the object that has to be shared and to make a description of the interface understandable by everyone. Now a extern software, in order to use this object, doesn't have to be interfaced directly with it, but it can use the XML interface that has been created for that object. If the data that has to be sent to the object is included in a XML structure based on the interface of that object, the object will receive the data in the right way and it will operate the calculation. This is easy to use and there are no problems due to incompatibility of platforms or of languages. The full description of the service is contained in a WSDL (Web Services Description Language) document and, inside of it, it's included a SOAP (Simple Object Access Protocol) message that describes the way to use that service.

## 2.2 XML and XML Schema

XML is a extensible and platform-independent way to structure data. It's not a programming language, but it's a set of rules that defines the way data can be structured. Its main purpose is to make possible a easy sharing of data across hosts with different platforms [6]. The data is structured in several elements, each element can have one or more attributes and can contain another element.

Every element is composed by two parts, both bracketed by a “<” and a “>”. The first one is the opening part and contains the name of the element and its attributes. The second part is the closing part and it contains only the name of the element preceded by a “/”.

```
<?xml version="1.0" ?>
<article title="article's title">
  <paragraph title="paragraph's title">
    <text>
      Text of the first paragraph of the article.
    </text>
  </paragraph>
</article>
```

If the element doesn't contain any other element, the second part can be skipped putting the “/” symbol at the end of the first part.

```
<element attribute='value' />
```

XML is extensible since it doesn't actually exist a tool of standard elements and standard attributes: every XML file has its own set elements and attributes. In the case above it has been used a element “article” with a attribute “title”: this element is not in a “standard element's tool”, but must anyway has been defined, as all the other elements, in somewhere. XML schema is a way to define the elements.

A XML schema is a XML file built to define a structure (schema) utilizable by other XML files, as the one above.

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="article">
    <xs:complexType>
      <xs:attribute name="title" type="xs:string" />
      <xs:element name="paragraph">
        <xs:complexType>
          <xs:attribute name="title" type="xs:string" />
          <xs:element name="text" type="xs:string"/>
        </xs:complexType>
      </xs:element>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

This above is a simple example that defines the structure used by the first file. There is defined a first element called article allowed to contain other elements (complexType) and a attribute of this element called title, this attribute must contain a string value. Inside the element article is defined another complexType element called paragraph, and so on. Infinite files, based on this schema, can be created. The schema simply defines a general structure that they must respect.

The thing that makes a WSDL document and a SOAP message understandable and utilizable by everyone, is that they're both based on a standard XML schema. There's a schema that defines how a WSDL document has to be structured and a schema that defines how a SOAP message has to be structured.

### 2.3 Conclusions

The aim of the present paper is to define a method to make different devices collaborate together autonomously. In this first chapter we saw what does collaboration mean and how to make a collaboration possible between different objects present on different devices. We also saw the problems that must be solved and the way they're solved through web services. Going back to the first example of the scientific laboratory and of the new device that has been bought, we can now understand better the situation. A method to enable these devices to collaborate together could be giving them one or more objects to share with the other devices. As we saw, it is not a easy task to solve the incompatibility of these objects, but a solution can be using something similar to web services.

## 3 Peer-to-Peer Collaboration

The word collaboration has now been defined and we saw how several devices can collaborate without any restriction due to any platform or language difference. Now is time to look at the second word: peers.

### 3.1 Peer-to-Peer Concept

The idea of a peer-to-peer network is the idea of a network completely decentralized, based on the collaboration of several nodes. Every node is autonomous and is able both to share and to utilize resources and services with the other nodes of the network. In a peer-to-peer network each node is then both client and server part and can provide, or require, a resource from another node without pass through a common "server". Because of this, theoretically, there isn't any central coordinating node of the network, but every node is directly connected to one or more other nodes (or peers). A central node can be the weak part of the network for some reasons:

- when the central node get broke, the net cannot work without anymore;
  - since all the network's traffic passes through it, it could become the bottleneck point of the network;
- In a peer-to-peer network there's no weak points:
- every node is useful to the network, but none is indispensable: the network can work even with without it;
  - there are not bottlenecks since the data traffic is well distributed in the network.

A peer is, then, a node able to communicate autonomously with the other nodes of the network, requiring and sharing data directly with them, without pass trough a central "coordinator" node.

If we consider the example of the scientific laboratory, we can suppose to create a simple peer-to-peer network in it, that enables a collaboration between three devices: a database server, a printer and the new testing apparatus that has just been bought. Each of these devices shares some services and it's connected to the network through a wireless connection.

The best choice would be to create the network through the Bluetooth technology, because of several characteristics it has. A Bluetooth device is able to:

- autonomously find every other active Bluetooth device near to itself;

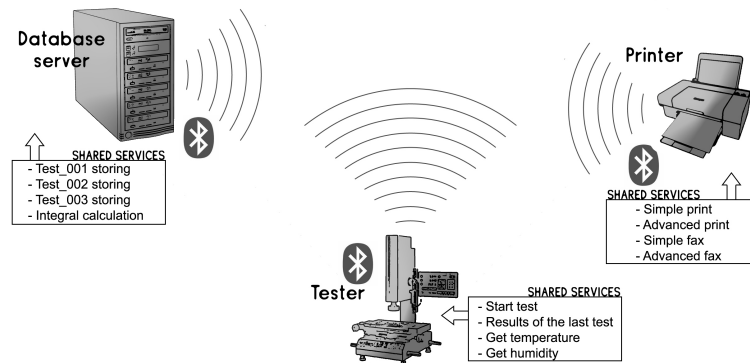


Fig. 1. Example of a simple Bluetooth network where devices are sharing some services.

- autonomously establish a connection with it (apart from the first time, when the connection can't be established automatically because a user needs to choose a common key and insert it in both the devices).

These three devices are then able to reach each other and to establish a direct connection each other. For instance, this could be a possible scenario (Fig. 1):

- the database server establish a connection with the tester and uses its Start test service;
- the tester starts the analysis after this remote request;
- at the end of the analysis, the tester establish a connection with the printer and uses the Simple print service,
- the printer prints out a sheet with the results of the analysis.

All of these operations has been performed through direct connections and every device managed every the request autonomously, just as in a peer-to-peer network should be.

### 3.2 Conclusions

A peer is exactly what a device, part of a APC network, needs to be. As said above, in a peer-to-peer network, every node:

- is autonomous,
- is useful but not indispensable,
- can be both client and server part,
- can directly reach every other node in the network.

These characteristics are exactly what an APC network needs. First of all, a device, in order to collaborate, needs both to use the services shared by the other devices and to be able to share its own services. When it uses a someone else's service needs to be the "client part", and when someone else uses a service of it, needs to be the "server part". In a client-server network, it isn't possible for a host to be able to become both server and client part, but it is possible in a peer-to-peer network. Furthermore, in a client-server network, a client is enabled to communicate only with the server. But what we need, in a APC network, is to let every host communicate with any other host of the network: this is possible in a peer-to-peer network.

## 4 Simple Peers Collaboration

A collaboration between two devices exists when at least one of them is able to share a service and the other one is able to use it. A peer-to-peer collaboration exists when both of these devices are able to establish a direct connection and both to share and to utilize a service. A autonomous peer-to-peer collaboration exists when these two devices can do these operations just by themselves, without any user intervention.

We defined what a collaboration is, which problems must be solved and how they can be solved. After this, we saw why a peer-to-peer network is the best way to manage a collaboration between devices. Now we still have to define the most specific part of the project: how can this peer-to-peer collaboration be autonomous? This is the most practical aspect and, in order to explain it, we need to look, step by step, at every operation computed by the devices. Let start to examine operations from the beginning using the same scenario as earlier. In this moment would be easier to explain, first of all, a model of a simple collaboration, where the user operates easy interventions to help the collaboration. Only in the next chapter we will have enough elements to define a model of an autonomous collaboration

### 4.1 Service Selection

The testing apparatus has been bought, everything is ready in the lab and the testing can start. After a while the first analysis, with the first type of substances, has been completed: now results needs to be elaborated.

First of all, what scientists needs, is to print results to store them in a paper archive. What they need is printer. Usually a printer needs to be installed on the device to be used by it, but it would be tricky to install a printer driver on a tester and to configure an application that works on the graphic part of the page that will be printed. Another solution might be to use the printer with a computer, but the results should be transferred in some way on the computer before to be printed. They don't need to do any of these operations because the devices are on an APC network. The scientist that is using the tester press the "select service" button in the menu of the tester's application: the tester will start to scan the area looking for some active Bluetooth device that shares some services (Fig. 3). Two devices has been found: one is a pc that works mainly as database server, and the other one is a printer.

Now that the tester has found these two devices, it sends them automatically a request for a list of the services they handles. As said in chapter 2, since they could be devices made in very different ways, it's impossible to them to understand each other without use something like web services. In this case, is needed a standard schema for the services lists. It doesn't matter in which way a device works, a XML schema can be managed by every kind of device. During the present paper some standard schema will be defined, this is the first one.

```
<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:complexType name="APC">

    <xsd:complexType name="service">
      <xsd:sequence>
        <xsd:element name="name" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:complexType>
</xsd:schema>
```

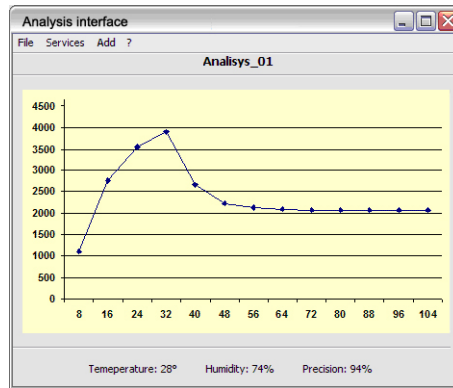


Fig. 2. The computed analysis, viewed on the graphical interface of the tester.

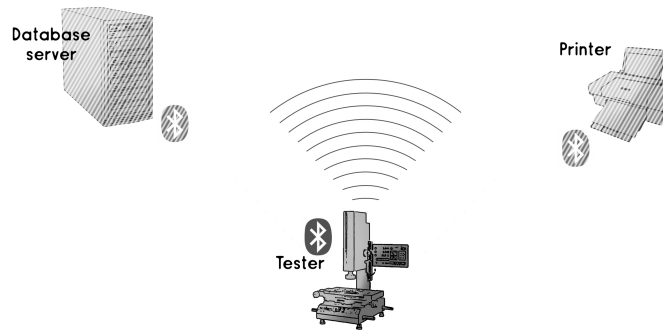


Fig. 3. The tester looks for some active Bluetooth devices.

```

        <xsd:element name="description" />
        <xsd:element name="devicename" />
    </xsd:sequence>
</xsd:complexType>

</xsd:complexType>
</xsd:schema>

```

In the definition of a services list may not be used this schema, this is just the basic one. But, in order to be readable by everyone, any other schema can only be built adding some features at this, but can't lack anyone of these elements.

Keeping the basic structure of the schema, we can now define the list of services of the printer.

```

<?xml version="1.0" encoding="UTF-8"?>

<APC-DESC:Envelope
xmlns:xs="http://www.w3.org/2001/XMLSchema"

```

```

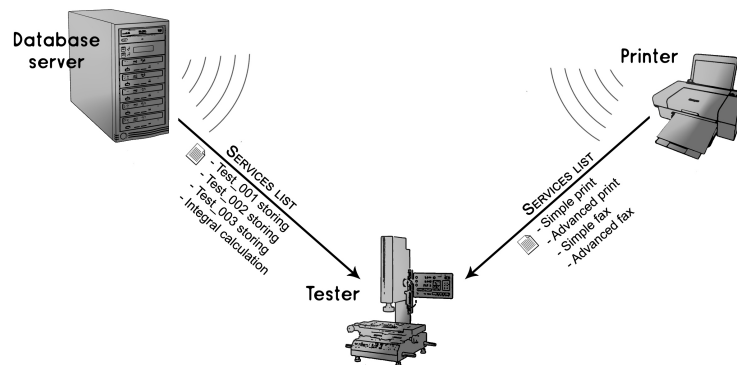
xmlns:APC-DESC="services-list-schema.xml">

<APC>
  <service>
    <name>Simple Print</name>
    <devicename>PRINTER-001</devicename>
    <description>
      Print of a simple text, without any kind of format.
    </description>
  </service>

  <service>
    <name>Advanced Print</name>
    <devicename>PRINTER-001</devicename>
    <description>
      Structured print with some possibilities to include
      author, date, title, subtitle, a graph and some comments.
    </description>
  </service>
...

```

Database server and printer send their services list to the tester (Fig. 4).



**Fig. 4.** The Database server and the Printer sends their services list to the tester.

The tester, knowing the services list schema, is also able to read the received lists. Once it has received them, it shows them on a user-friendly interface (Fig. 5), so the user (the scientist) can be able to choose which service wants to use.

In this case is the Advanced print service of the printer001 seems to be the most proper service. The scientist chooses this service. The service has been chose, now it needs to be used.

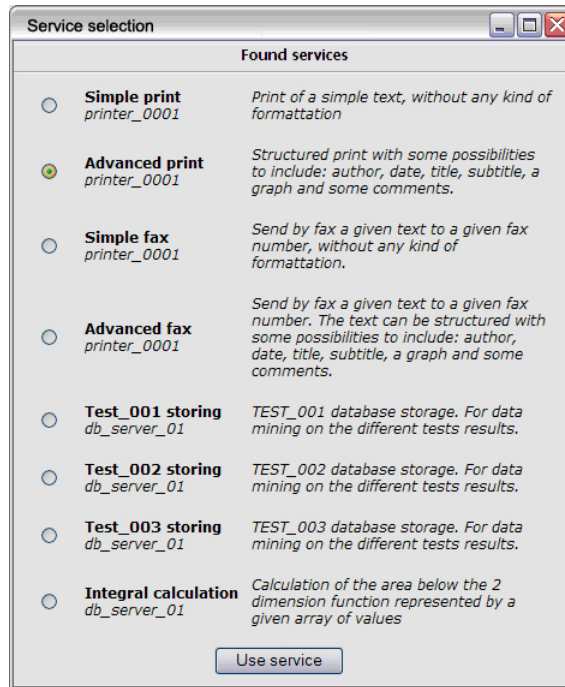


Fig. 5. The services lists received by the tester, viewed on its graphical interface.

## 4.2 Service Utilization

How can the tester know how to use this service? It can't know it now, but it will learn soon. What is needed, first of all, is a detailed description of the service, like a "user guide" to the service.

The tester opens a connection with the printer and sends it the request to receive details about the Advanced print service. As always, the request can't be understood by the other device if this is not made in a platform-independent way, and, once again, the solution can be found using a standard XML schema. As for the services list, there must be a basic XML schema that defines the standard outline of any service details request message.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:complexType name="APC">
    <xsd:complexType name="service-details">
      <xsd:element name="name" />
    </xsd:complexType>
  </xsd:complexType>
</xsd:schema>
```



As the services list one, this schema is just a basic outline. A service details request message is allowed to contain other elements and attributes. What matters, in order to make it compatible with every device, is that it shouldn't lack anyone of the basic elements. Keeping, once again, only the basic schema, this is how the tester's request looks like.

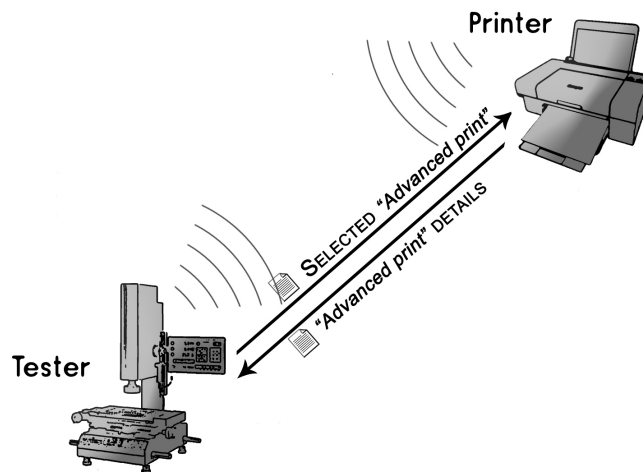
```
<?xml version="1.0" encoding="UTF-8"?>

<APC-DESC:Envelope
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:APC-DESC="details-list-schema.xml">

  <APC>
    <service-details>
      <name>Advanced print</name>
    </service-details>
  </APC>

</APC-DESC:Envelope>
```

Once the printer has received and understood the message, it sends the details back to the tester (Fig. 6).



**Fig. 6.** The Tester asks details for the Advanced print service and the Printer replies.

A third standard needs to be defined, to make the tester able to understand the service details message sent by the printer.

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```

<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:complexType name="APC">

    <xsd:complexType name="service-details">
      <xsd:element name="input" />
      <xsd:element name="output" />
      <xsd:complexType name="description">
        <xsd:element name="input" />
        <xsd:element name="output" />
      </xsd:complexType>
    </xsd:complexType>

  </xsd:complexType>
</xsd:schema>

```

The Advanced print service details message, based on the basic schema, looks like this.

```

<?xml version="1.0" encoding="UTF-8" ?>

<service-details>
  <input>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:complexType name="Headline">
        <xs:choice>
          <xs:attribute name="Author" type="xs:string" />
          <xs:attribute name="Date" type="xs:boolean" />
        </xs:choice>
      </xs:complexType>

      <xs:complexType name="Header">
        <xs:choice>
          <xs:attribute name="Title" type="xs:string" />
          <xs:attribute name="Subtitle" type="xs:string" />
        </xs:choice>
      </xs:complexType>
    </xs:schema>
  </input>
  <output>
    ...
  </output>
  <documentation>
    <input>
      #Headline: "These elements will be printed on the
                  top band of every sheet."
      #Author: "Will be in the top-right corner of every
                printed page."
      #Date: "Will be in the top-right corner of every
              printed page, under the Author."
    </input>
  </documentation>

```

```

<output>
...
</output>
</documentation>
</service-details>

```

The structure is basically composed by three main complexType elements: input, output and documentation. The documentation element contains another input and another output element. The first two elements defines two schema:

- Inside the input element is defined the XML schema of the service's input interface,
- Inside the output element is defined the schema of the service output.

The third element contains the description of every part of the input and output schemas. The printer sends the service details message and the tester receives it. Once the message is arrived, it's shown on a form (Fig. 7), in the tester screen.

"Advanced print" service		
<b>Headline</b> <i>These elements will be printed on the top band of every sheet.</i>		
Author	Will be in the top-right corner of every printed page.	
Date	Will be in the top-right corner of every printed page, under the Author.	yes
<b>Header</b> <i>These elements will be printed at the beginning of the first sheet.</i>		
Title	Will be in the top-right corner of every printed page.	ANALYSIS_01
Subtitle	Will be printed with a strong font centred under the Title.	Sodium: 23,48 / Potassium: 41,02 / Carbon monoxide: 16,06
<b>Body</b> <i>These elements will be the main part of the print.</i>		
Text	Will be printed beginning under the Subtitle, in a normal style, aligned on the left, and, if needed, will be split in several sheets.	time_slot: 0,8 value: 1102 time_slot: 1,6 value: 2763 time_slot: 2,4 value: 3546 time_slot: 3,2 value: 3913 [...]
Graph	<i>The graph will appear at the end of the text.</i>	
Graph_x time_slots	This field must contain an array of values that will be used for the x axis. Values can use decimal digits.	0,8 1,6 2,4 3,2 4,0 4,8 5,6 6,4 7,2 8,0 8,8 9,6 10,4
Graph_y values	This field must contain an array of values that will be used for the y axis. Values can use decimal digits.	1102 2763 3546 3913 2673 2229 2135 2087 2073 2069 2069 2070 2069

**Fig. 7.** A form, on the graphical interface of the tester, to let the user interface the analysis' data with the service.

Now the scientist has the "user guide" to the service and is finally able to use it. The form must show at least two fields for every element of the input and output schema of the service:

- The description of the element,
- A list of values that can be associated at this element.

While the content of the first field is supplied by the printer in the service details message, the list of values of the second field must be supplied directly by the tester. There might be several ways to create this list. An easy way is to structure the data of the analysis in a XML file. In the next chapter will be more clear the convenience of this method.

The form is then created dynamically on these two XML structures and the scientist can now interface the analysis data with the service:

- The author camp is to be left empty.
- The date has to be printed.
- The tile is the test name (in this case is "Analysis-01").
- The subtitle is the array of used substances.
- ...

When the "OK" button is pressed, a XML file will be created using the form's contents. This means that the XML file will be based on the Advanced print schema, supplied in the service details message, and will contain the test results in the way that the user has chose.

```
<?xml version="1.0" encoding="UTF-8" ?>

<Headline date="true">

<Header Title="Analysis-01" Subtitle="Sodium: 23,48 / Potassium: 41,02 / Carbon monoxide: 16,06">

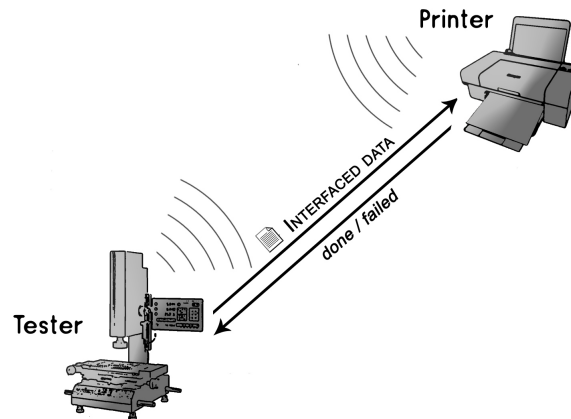
<Body>
  <Text>
    timeslot: 0,8 value: 1102 / timeslot: 1,6 value: 2763 /
    timeslot: 2,4 value: 3546 / timeslot: 3,2 value: 3913 /
    timeslot: 4,0 value: 2673 / timeslot: 4,8 value: 2229 /
    timeslot: 5,6 value: 2135 / timeslot: 6,4 value: 2087 /
    timeslot: 7,2 value: 2073 / timeslot: 8,0 value: 2069 /
    timeslot: 8,8 value: 2069 / timeslot: 9,6 value: 2070 /
    timeslot: 10,4 value: 2069
  </Text>
  ...

```

The tester sends the created message to the printer (Fig. 8). The printer will now understand it and it prints a sheet with the sent data. In the end, the printer will send in output a XML message back to the tester. The tester is able to understand it because the schema of this message has been sent together with the service details message, inside the output element.

### 4.3 Conclusions

A new testing apparatus has been bought, the scientist executed the first test with it and he wanted to print the results. He pressed a button from the software of the tester and it found all the APC services shared by the neighbour devices. The scientist has selected a service and has easily interfaced the test data with the input schema of the service. The tester has composed the request and a sheet, with the wanted data, has been printed. Nothing has been installed or configured; nothing needed some kind of special compatibility or new updates to work properly. The scientist has now a



**Fig. 8.** The Tester sends the interfaced data to the Printer, and the printer returns if the printing process was successful or not.

powerful tool that gives him a lot of new possibilities. He can easily use every service shared by the other devices of the lab. This is a good result, but the system is still non-autonomous; this is just a simple peers collaboration. The solution though is near; the compatibility problem between the devices has been solved and a lot of operations are yet managed autonomously by themselves. The only problem left to solve is the user intervention in the:

- Selection of the service,
- Data interfacing with the service.

## 5 Autonomous Peers Collaboration

Finally it's the time to define the word autonomous. An entity is called autonomous when it computes a job independently, without the intervention of any other external entity. In this specific case, it means that the collaboration, between two devices, must work without any external intervention. In the simple peers collaboration we saw how it is possible to manage, in a simple way, collaboration. The user external intervention is reduced at the minimum, but how can this collaboration become completely autonomous? The solution is now easy. Basically, the two things that the tester is not able to do autonomously, are the same things that a person, without any experience, wouldn't be able to do as well. In the same way, as a person, gaining some experience, becomes able to choose the service and to interface it, also a device, gaining some experience, will be able to operate these operations autonomously. Everything depends on this, but what does "gain some experience" actually mean? I gain experience when someone teaches me something and I try to do it by myself. For instance, let suppose that I have to go in a place where I have never been: I have no "experience" in doing this. The first time I go there, I don't know how to reach that place, then I need someone to teach me how to go there. But, after this time, I will always know how to get to that place. Why? What happened? I simply memorized the route in my mind. From this moment on, every time I need to go to that place, I will always remember

the route and I will go there autonomously. In the same way, the tester didn't know which service select, and neither how to interface the analysis data with that service. It couldn't know it because it was the first time that it performed this. But, if the tester becomes able to memorize (store) these operations, after the first time it will always be able to remember them and work on this job autonomously.

More specifically, the data that the software of the tester needs to store are:

- The name of the service and the name of the device that shares it,
- The "way to interface" the analysis data with the service.

The first information is easy to store, because is composed just by two names (for instance printer-0001 and Advanced print). The second one is harder to store, since is not composed by static values like the first one, but is composed by references to parts of the analysis results. For instance, in the present example, the tester can not only memorize that text element of the Advanced print input interface must contain: "timeslot: 0,8 value: 1102 / timeslot: 1,6 value: 2763 / timeslot: 2,4 value: 3546 / ...". These are the results of the analysis and every time they changes. If it stores just this information, every time the text element will be the same. This is not what we need. The tester needs to memorize, instead, that the text element must contain the analysis results, expressed in that kind of text structure. If it stores this kind of information, the text element will contain every time the new values of the analysis results. There could be more than one way to do this, and every device can use a different one, but in this paper I want to suggest a specific method. This method is based on a XML structure, like most of the things are in this project.

## 5.1 Stored Automation Interfaces

In the chapter 4.2, it was suggested to insert the analysis data in a structure, as a method to let the user easily interface the analysis data with the service. That same results structure comes in useful in this case too.

Analysis data structure:

```
<?xml version="1.0" encoding="UTF-8" ?>
  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="Analisis-01" xmlns:tns="
<APC>
  <name>Analisis-01</name>
  <substances>
    <substances-txt>Subst-01: 30,00 ml / Acetic acid: 50,00 ml / Ammonia: 41,02 ml</substances-txt>
    <substance quantity="30.00">Subst-01</substance>
    <substance quantity="50.00">Acetic acid</substance>
    <substance quantity="41.02">Ammonia</substance>
  </substances>
  <results>
    <results-txt>
      timeslot: 0,8 value: 1102 / timeslot: 1,6 value: 2763 /
      timeslot: 2,4 value: 3546 / timeslot: 3,2 value: 3913 /
      timeslot: 4,0 value: 2673 / timeslot: 4,8 value: 2229 /
      timeslot: 5,6 value: 2135 / timeslot: 6,4 value: 2087 /
      timeslot: 7,2 value: 2073 / timeslot: 8,0 value: 2069 /
      timeslot: 8,8 value: 2069 / timeslot: 9,6 value: 2070 /
      timeslot: 10,4 value: 2069</results-txt>
    <timeslots unit="second">
```

```

    0.8 1.6 2.4 3.2 4.0 4.8 5.6 6.4 7.2 8.0 8.8 9.6 10.4
  </timeslots>
  <values>
    1102 2763 3546 3913 2673 2229 2135 2087 2073 2069 2069 2070 2069
  </values>
  <result time="0.8">1102</result>
  <result time="1.6">2763</result>
...

```

The structure should store the data in different ways in order to be more flexible and adaptable to different situations. In the present situation, for instance, the scientist selected to associate the substance-txt element, of the analysis structure, to the subtitle attribute, of the Header element of the Advanced print input interface. Substances have been stored in 2 different ways:

- substance-txt element,
- substance elements.

If they were stored only in the substance elements, it wouldn't have been possible to print the subtitle in that way.

Every time that a new analysis is computed, the data of the analysis is structured with the same schema as earlier. Using the references to the elements of that schema, we can store a general way to interface the data of every test with the service. The resulting file is an automation interface. This automation interface will be an association between the analysis structure and the Advanced print input interface.

```

<?xml version="1.0" ?>

<Headline Date="true" />

<Header Title=#APC/name# Subtitle=#APC/substances/substances-txt#/>

<Body>
  <Text>#APC/results/results-txt#</Text>
  <Graph>
    <Graph-x>#APC/results/timeslots#</Graph-x>
    <Graph-y>#APC/results/values#</Graph-y>
  </Graph>
</Body>
...

```

The file is based on the structure of the Advanced print input interface. The value of every element and every attribute could be:

- A static value (like the attribute Date that must have a Boolean value),
- A dynamic value, which is the reference to an element of the analysis structure, bracketed by two hash signs.

Once that the tester stores this file, it will always be able to compute this job autonomously.

In order to be completely autonomous, the tester needs to associate this automation interface to the right event: the finishing of an analysis.

## 5.2 The first autonomous collaboration of the Tester

If a second analysis is computed, these will be the operations step by step:

- the scientist executes the analysis with the testing apparatus,
- at the end of it, the tester structures the resulting data through its own schema,
- the tester looks for any stored interface associated to that event (the finishing of the analysis) and it finds the stored interface of the Advanced print,
- it replaces every value tagged by the hash signs with their corresponding values in the analysis structure,
- it opens a connection with printer-0001,
- it sends the created XML file to printer-0001,
- the printer prints the results and it sends a confirmation back to the tester.

## 5.3 Conclusions

The job is now autonomous. In every test that will be executed this operation will now be done automatically. This was just an example with only one stored interface, but it's unlikely that a scientific test could be this simple: it will probably be composed by a lot of operations that has to be executed, every time, on several devices. With the APC technology, one or more operations can be associated to every event of every device. The devices will execute all of these jobs every time autonomously, without the intervention of anyone.

# 6 Additional Analysis

## 6.1 Device-oriented programming

The first concept that we saw in this paper (chapter 2.1), was the concept of “object-oriented programming”. This is the first concept of collaboration that we saw, but we finished with it when a serious problem has been discovered in it: the incompatibility between objects developed in different languages or in different platforms. The object-oriented topic has been dropped, but an APC network is still using something very similar to objects: devices. We defined that an object as an entity that has a role in a computer program and is able to:

- share some functions with the other objects,
- utilize, autonomously, functions shared by other objects.

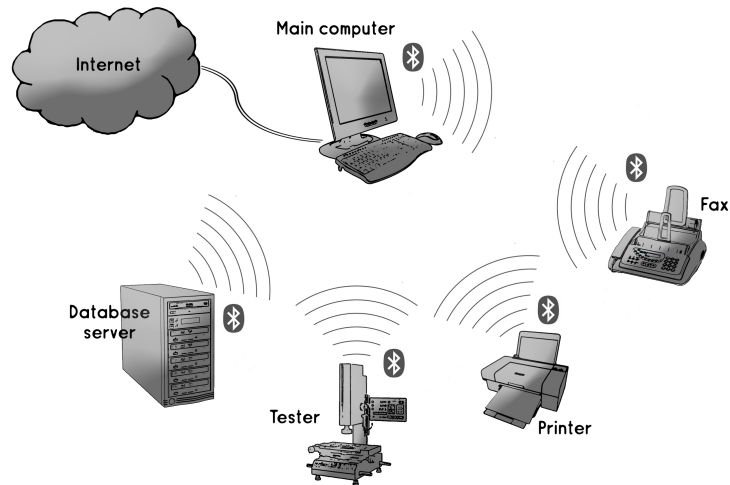
A device, part of an APC network, possesses the same proprieties in relation with the other devices of the network. Therefore, a device, part of an APC network, can be considered like an object. In the same way, the services shared by that device, can be considered as the public functions of an object.

From this point of view, the devices of an APC network can be also managed as objects and we can find a way to use them as the objects are used in an object-oriented language. We just need to define a main function, as it is in object-oriented software. The main function should be managed by a computer, part of the APC network.

## 6.2 Data security

Everything in this project aims to be the most open as possible. Therefore everything tries to be the most simply as possible, any addition can be a barrier that compromises the compatibility between the devices or even some new way to use the network.





**Fig. 9.** An APC network and a computer, part of it, that can be use the network through a device-oriented software.

Because of this, there's no implementations of some kind of data secure management; somehow, if the network is based on the Bluetooth technology, this doesn't really matter. Bluetooth uses a protected channel to send data. A connection needs a key, which is common for every device that takes part of the transmissions. If the network technology used it's not Bluetooth, the data secure management has, anyway, to be implemented in the network layer, leaving the higher layers free from any kind of new implementations. This would be less dangerous to make the network not compatible with every kind of device.

### 6.3 Hardware implementation

Looking at the project, one of the first things that can be considered, is that all of these devices seem to need to be very technological, with a Bluetooth connection, an operative system, an applicative layer and a HTTP browser that handles XML files. It is quite expensive to have something like this on every device of the network; but that's not actually needed. Basically, what is a XML file? Nothing more than a big array of ones and zeros. Furthermore, what is a service if not just an operation computed by the device, described by a XML file and that gives as result another XML file? Everything is basically just an array of ones and zeros. Because of this a device, in order to be able to share a APC service, it doesn't really need a screen, a keyboard, a SO and some applications compatible with XML, but it can work just with a Bluetooth board, a small memory and a assembly-programmed component. For instance, the printer of the scientific lab, it doesn't need to have a screen and a SO, but it can even have just a hardware implemented APC technology that emulates a well implemented one.

## 6.4 With a Bit of Imagination

In this paper, I outlined a theoretical way to create an autonomous collaboration between different devices, compatible with every kind of platform. I talked just about a scientific environment, but this is not the only field where an APC technology can be used. With a bit of imagination this technology can be used in a lot of different ways. As saw above, a further application of APC can be to create device-oriented software; this is just another example. Another application can be, for instance, to use an APC network between the household appliances. Let suppose that your old mobile phone get broke and you buy a new one. It doesn't matter the producer, or the software it has. When you're back home with your new mobile phone, you can easily tell it to print any text you'll receive, or you can even use it to tell the heating system to reach a established temperature. If your oven has this APC technology, you can even tell it to, automatically, send an advice to your mobile every time that the time of cooking is up. Several more applications can be created. APC technology is just a base that can be used in a much opened way, in a lot of different fields, by several kinds of application.

## References

1. Rosenberg, R.: The Social Impact of Computers. Elsevier Academic Press (2004)
2. Sun Microsystems, Collabnet: Jxta project (2001) [Http://www.jxta.org](http://www.jxta.org).
3. Simpson, J., Weiner, E.: Oxford English Dictionary (second edition). Oxford University Press (1989)
4. Meyer, B.: Object-Oriented Software Construction. Prentice Hall (1997)
5. Lindholm, T., Yellin, F.: The Java virtual machine specification (second edition). Prentice Hall (1999)
6. Tim Bray, Jean Paoli, C.M. Sperberg-McQueen, Eve Maler, François Yergeau: Extensible Markup Language (XML) 1.0 (fourth edition) - Origin and Goals. W3C (2006)

# Textual Advertisement Models—A Comparative Look

Abubakr Saeed

Department of Computing Science  
Umeå University, Sweden  
ens03asd@cs.umu.se

**Abstract.** The computer has provided the monetized search, called “paid listing model” which revolutionized the business world. The paid listing model uses sponsored search engine lists to attract potential customers to a particular web site. In paid search, content providers pay search engines to show sponsored links in response to user queries beside the non-sponsored links. Overture is credited with developing the paid listing model by 1998, which now provides paid listings to the three largest portals: Yahoo, MSN and AOL. Google introduced his own paid listing model, AdWords. Today these two firms capture the majority of paid search traffic. This paper is aimed at increasing the general understanding about paid listing. The opening part highlights the search engine market place and presents listing techniques. In next section, I attempt to compare the foremost contextual advisement models. The ending piece covers dilemma of search engine marketing, from the advertiser’s and the user’s perspective.

## 1 Introduction

The ever advance capability of search engines has diminished the gap between people and information. Search engines provide wider access to information and fetch the most related one from the heap of information on the web. The basic operation of search engines is to scan the text on pages with the help of some “indexing program” and then rank these pages with help of algorithms. Early on, this fundamental functionality was confronted with a scam, get a high rank by providing repeated keywords in a web page. Search engines adopted new mechanisms, i.e. “human edited directories (Yahoo)”, “off the page factors” and “ranking on the basis of most accessed links (Google: Page Rank)”.

Later an inventive concept, “interspersing result listings with advertisements” was introduced by GoTo (Overture), in which an open auction of keywords was developed as the primary mean to rank the page [1]. This idea was named “Pay Per Click” and currently most search engines follow this mechanism.

This paper introduces search engine marketing before outlining a number of paid listing techniques. Afterwards, the few leading models will be discussed in detail with their methodologies, features and their business perspective. The remainder of the paper explains search engine marketing concepts from the advertiser’s and the user’s viewpoint.

## 2 Search engines listing

Search engine marketing fits in the category of “pull marketing strategies” [2] and has distinctive traits. Customers see the ad when they want to and an ad can appear multiple times but the advertiser has to pay once or every time it clicked. According to the search engine marketing report [1] costs on search engine marketing totalled 9.40 billion last year, up 62% from 2005. Search engines perform the marketing through two distinct sorts of listing: “organic listing” and “paid listing”, the major contributor to search engines marketing revenue is paid listing.

The screenshot shows a Google search results page for the keyword "Air ticket". The search bar at the top contains "Air ticket" and the search button is labeled "Sök". The search results are displayed in a grid format. The top row of results is highlighted in yellow and contains sponsored links. The first sponsored link is "Billiga Flygbiljetter" from Seat24, with a "Sponsrade länkar" label. The second sponsored link is "Air Ticket" from supersavertravel.se, also with a "Sponsrade länkar" label. The third sponsored link is "Billiga Flygbiljetter" from flygvaruhuset.se, with a "Sponsrade länkar" label. The right side of the results contains several more sponsored links, including "Air Spanair", "Air Ticket" from Lufthansa, "Jämför Flygpriser Online", "Airlines Ticket", "Cheap Air Fares", "Cheap Worldwide Flights", and "Last Minute Air Ticket". Below the sponsored links, there are organic search results, including "Cheap Airline Tickets – book discount airfares at Cheapflights.com", "Cheap Airline Tickets, Discount Hotels and Airfare - SideStep", "Northwest Airlines - Airline Tickets, Plane Tickets & Airfare", and "Airline Tickets, Cheap Flights, Cheap Plane Tickets, Hotels, Car ...".

Fig. 1. Result screen of keyword “Air ticket”, sponsored links are on top and right side

### 2.1 Organic listing

Organic lists are also called natural lists. These lists are considered as non-biased and most reliable. Organic lists are made on user’s query by the search engine’s

matching algorithm. Algorithm searches the typed query in the text of available website database and ranks the list according to more related information. Web sites require optimization for getting the high rank in the list, especially in the content and navigation. Organic lists bestow great extent of credibility and extensive exposure on all search engines.

## 2.2 Paid listing

In paid listing, search engine sells the “keywords” to the advertisers. This is also called “pay per click” (PPC), appears at the top and the right side of search results. Normally it work like this: advertisers choose a list of keywords that they think potential customer might be interested in, then they agree to pay for each click (through bid) that the user does through the search engine. If the user views that search website and clicks on that particular advertisement’s link then advertiser has to pay some predetermined money. Paid listing is considered as rapid results producer, no need to optimize the website and be guarantee to be in the list at desired rank.

## 3 Search engine listing techniques

According to [3] nearly all leading search engines make their lists by the crawling or the web directories method. Crawling based search engines use a pragmatic module that gives instructions to “spider” that includes: a set of URLs to visit and scan, a starting link, the following path and a stopping point. A Spider maintains a record of their travel path in the form of massive large data, named as “indexing”. For instance, “search.ch” search engine crawls only the Swiss web pages and have limitation of geographical borders of Switzerland. Irrespective of search engines based on web directories, crawling search engines dynamically update themselves. Search engines make their lists (organic and paid) by matching “search query” within an available indexing and rank them by relevancy algorithms (organic). In an organic or a natural listing, the web site owners can’t influence the ranking algorithm to get a higher ranking. Consequently, they try to optimize their websites in particular manner to be ensure of their web pages accessibility to the search engines by following the working of ranking algorithms.

For paid listing, search engines mainly use three techniques namely [3]:

### 3.1 Paid inclusion

Paid inclusion is the simplest form of income generator. In this technique site owner (advertisers) are charged to guarantee that their new site will be included in a search engine or directory’s listing soon but it has no concern with the ranking, page can appear at low ranked for any specific search. The services are commonly charged by per link (URL) basis and provide inclusion for a year. Paid inclusion appears to the user as an organic list rather than paid list. The Inktomi, AltaVista, and Yahoo directories have dominance in this technique.

### 3.2 Contextual distribution

Search engines also distribute paid-ads to their network partners (search tools, websites and media outlets) on the basis of keywords found in the text of these sources. In this technique, the network partners display desired received advertisements from the search engines, on their sites, on the basis of keywords and phrases found in the text the ad is placed beside. Many famous media outlets AOL, MSN, NY Times, Seattle Times, National geographic, CNN, Toronto Star, Knight Rider News etc. [4] show the advertisements, taken from different search engines.

In contextual distribution, another different schema for generating revenue is to sell profile data to various companies. As the search engines collect massive amount of user data on the daily basis and they use it for search engine optimization but beside they sell it to the companies who use it for commercial needs.

### 3.3 Paid-placement

Paid placement is a keyword driven mechanism which guarantees ranking positions to the advertisers. Advertisers choose a “keyword or phrase”, which might be attractive to their customers, and bid against it. In general, the higher bid gives the higher ranking in the list. Simply, for receiving top ranking of a site, pay more than other advertisers. Paid lists make up in order from the highest bid to the lowest bid. The services are charged once by the bid then pay for each click on advertised link through search engine. That’s why, advertisers chose most targeted keywords that attract potential customers. Paid placements also named as “Pay for Placement” and it appears to users as “Features Listings” or “Sponsored Links”. The leading companies in this area are Google and Overture/Yahoo.

An added trait with this system is the distribution of the advertisement to other sources, in form of contextual distribution. Almost all paid placement programs perform in this fashion. Paid listing is a cost effective mechanism especially for small and new enterprises, which provides them a possibility to compete with the large competitors.

## 4 Comparison of models

Search engines are primary tool for helping users to find relevant information on the web. According to [5], the search engines evolved to a viable mass medium tool, when internet’s backbone in USA was privatized and the internet started to use primed for the advertising. By 1998, there were five leading search engines: Google, Alta Vista, AlltheWeb, Teoma, and Inktomi. The search engine industry acquired a breakthrough when Goto.com introduced a usage of its organic algorithmic searches with a database of advertisers and named as “Pay-per-click method”. Later in 2000, GoTo renamed as “Overture”, focused on its robust aspect: commercial advertising database and by 2002, number of advertisers reached up to 80,000.

Inktomi was the first to introduce concept of paid inclusion by providing the guaranty to include the web address in the organic algorithmic searches. This concept provided the gateway for search engines to find massive monetary benefits.

Google started its text advertisement campaign in 2002 with its model, AdWords and within three years Google's advertising profit alone grew nearly 500 percent, estimated revenue is 1 billion by 2003, and persists to surpass the market expectations [6, 7].

For gaining a greater market share, Yahoo acquired Inktomi, Overture purchased Allthe Web and Alta Vista and later Yahoo took over Overture and emerged as "Yahoo! Search Marketing". Thus, Yahoo swallowed three top search engine providers and stayed alone in the market to compete Google. Yahoo/overture and Google/AdWords have following bidding techniques and strategies [8]:

#### 4.1 Overture / Yahoo

As Overture had initiated a twist "Pay per Click method (PPC)", in which a highest bid is selected from advertisers against a specific keyword and the advertiser has to pay initial bid amount plus a particular sum on each click on the link by user. In PPC method advertiser who places the highest bid shows at the top of the sponsored list, with other advertisers shown below in the reverse bid order.

- Listing is purely based on the bid amount, higher bid gets higher ranking.
- Overture has a fixed bid mechanism. It allows an advertiser to set the exact amount they are willing to pay for each click. In the fixed bidding advertisers pay the precise sum they input, regardless of what the other advertisers in that marketplace bid. Later overture started the auto bidding system as AdWords has.
- Typical bids range from \$0.05 to \$3.00, but it varies. The link between bid and ranking is normally non-linear. Overture had consistently raised their minimum bid (reserve price), from \$0.02 to \$0.10 over two years. Advertisers can bid only in native currency of the country, where they want to advertise. This makes indifference of minimum charges across countries. For instance: minimum charges for ad shown in USA (\$0.10) and in UK (£0.10) are not same.
- Overture is currently running with an editorial backlog and advertisement's relevancy is checked by editorial staff before approval, the keyword approval can be up to a couple of days.
- Overture provides the facility to advertisers, to know the conversion rate of their advertisements (it is a ratio between users who visited ad, made purchase) and calculate their Return on Input (ROI).
- Overture displays correct spelled advertisements, misspelled advertisements are automatically corrected. Overture offers exact match, groups phrase match and broad match.

- Any word that you do not want your ad to show for can be blocked by a “negative keyword” option. AdWords also supports this option.
- Overture provides PPC facilities to main search engines like: MSN, Yahoo, Netscape, InfoSpace, AlltheWeb, Alta Vista and Lycos/Hot Bot. Content sites include: CNN, Advertising.com, CitySearch, National Geographic, Wall Street Journal and others

## 4.2 Google: AdWords

Google has been involved in the search technology from the company’s inception (1998) and it brought AdWords with the more innovative mechanism. AdWord’s listings are ranked not only by bid amount but also on Click through rate (CTR). CTR is an approach of measuring the success of displayed advertising, it calculates by dividing the number of users who clicked on an advertisement on a web page by the total number of times the advertisement was shown.

- AdWord prioritize CTR, respect to bid amount. Therefore advertiser with a high CTR and low bid gets a high position in the list than a competing advertiser with high bid with low CTR.
- Another innovation that makes AdWords mechanism different is: “auto bidding”, in which the price actually paid for the click is lowered based on how high the competition is. The advertiser only pays \$0.01 more than the maximum bid of the next highest competitor. This feature is also named as “Adwords Select Discounter”.
- Advertisers are provided by an advance keyword matching options, through this feature they could be more precise in their search keywords.
- AdWords advertise the advertisements on the search engine without checking the relevancy but the advertisements are make sure for their relevancy before syndicating to other search partners.
- No restriction of the language or the territorial boundaries.
- Low prices provided a lower \$0.0511 minimum in general and the bids are allowed to be placed in any currency. In this way AdWords keeps its minimum charges same in any country.
- AdWords does not provide spelling auto correction facility like Overture does.
- AdWords provide PPC facilities to AOL (USA), AskJeeves, Netscape, and Earthlink. Content sites include: NY times, Forbes, ABC, US News, Fox, Business world, Weather Channel, and others.
- Advertisers can limit their daily budget total, once the budgeted sum is over, the ad is dropped from the list.

AdWords and Overture, leaders of the contextual advertisement market with a total of 90% of the market and are dominated this marketing world for the last two years and they are progressing aggressively. The remaining search providers can either maintain to syndicate PPC results from market leading suppliers or they can develop (or acquire) their own PPC mechanism.



In the end of 2006, Google has extended AdWords to AdSense, which allows advertiser to bid not only on specific site but also on AdSense channels. For covering new channels Google has purchased YouTube for \$1.65 billion for video distribution. Google has started a new channel “pay per call” for the targeted industries, for instance real state, etc. [9]

## 5 Browser’s Ad-blockers

Today’s customers are capable to filter out information which they do not wish to receive, they decide what they want to see by the help of diverse technologies, such as ad-blockers, Spam filters and RSS feeds. There are already plug-ins for web browsers for instance Internet Explorer, Firefox and Opera that block the ads by the search engines. For additional filtration of received material there were banner blockers and pop-ups blockers. As in contextual advertising search engines distribute their lists through contextually relevant placements on content sites and these content sites often display undesired ads. In this respect, different software also blocks the paid search listings and contextual ads.

Blocking ads is not a straightforward task because business community allows them to operate openly to attract their customers. Further, Unwanted ads can appear in all sizes and forms therefore utilizing a single technique against them are not enough to stop them. But in general, ads are used to block by not allowing an unauthorized process to access certain URL. Operating systems have built-in capabilities to stop the pop-up ads, in addition service packs enhance the capabilities of operating system to protect the user from unwanted ads.

## 6 Dilemma of search engine marketing

In the search engine marketing sphere, there are certain participants, who are directly tied with the strategies of search result providers: search engine’s user and advertiser. Search engine user concerns with the search results that best match their queries and the advertiser desire to have a best return to their input.

### 6.1 From searcher’s perspective

Research studies [10] show that common search engine user has little knowledge of search engine’s inside functioning for lists. They have belief that the search engines show unbiased results on the first page. Further they prefer the organic lists compared to sponsored lists and consider organic results before seeing sponsored results. In [11] study they shows that 82% searchers start with the organic results.

As paid listing has turned into a booming business for its providers, this economical margin tempts them to provide more contextual advertising and less non-paid results. Generally, ads are appeared under the labelled “sponsored

listings” but they place within the “organic or real” listing that is considered as unbiased. The problem is searchers ought to receive paid results even in organic listings. Study [12] examines the resemblance in organic and paid results and found the same query result in eight different search tools. That’s mean same paid results are listed in almost all as the top-ranked results without labelling them as “sponsored”. Searchers trust search engines to present only real results in the organic lists but they have no idea that this is make up by the top paid advertisers. In this way, search engines might be losing their credibility.

Both Google and overture are turning more like online ads agencies rather than to be search engines and like ads agencies they collect consumer data (by pixel tags, cookies and contact/personally identifying information ) to increase ads performance. All this kind of user data is valuable to marketers. Consequently, user feels insecure regarding their privacy.

## 6.2 From advertiser’s perspective

Survival of fittest, advertisers who can bid high will be at top rank in the list. Advertisers with a limited resources can never compete with advertisers with bulky resources, therefore they go for getting rank in organic listings. Further, as immature advertisers entered the online marketing, in order to get a high rank they place ridiculously high bids and imbalance the particular “keyword” campaign.

Advertisers face two types of click frauds, first called “competitive click fraud” when invalid clicks on ad occurs by competitor without having actual interest in a targeted ad and advertiser have to pay money to a search engine for each click on their link. Often competitors play unclean by exhausting a rival’s pay-per-click advertising budget. Second, “network click fraud” occurs when an ad-network partner (who receives ads in form of paid placement from search engines, displays them on their end and charge against each click on ad) generate fake click traffic for earning more from syndicating search engine.

It is the responsibility of the search engine to protect advertisers from the click fraud. As search engines have not done enough to restrain click fraud, as a result, at least two lawsuits have been filed against Yahoo! and Google by advertisers.

Search engines use various click measurement tools to analysis the traffic for stopping click fraud. For instance, filtering duplicate clicks generated from the same computer, signature-based methods (on advertiser-side) perform tracking of cumulative data to get expected behaviors. Since there is no particular criterion for what constitutes a valid click, therefore it is hard to overcome this crisis [13].

## 7 Conclusion

Internet advertisement marketing particularly PPC has developed extensively, and turning over an estimated \$1.1 billion per year and still growing at 13 percent

per quarter. The average user has little knowledge of search engine's lists: how these lists are ranked? In an attempt to increase the understanding of general user, this report takes an impact on the online market place and focuses on the paid listing. In addition, two leading textual advertisement models, AdWords and Overture are relatively conversed.

Definitely the objective of search engines is to acquire the right information to the user in right form. Competition between search engines drive them to enhance their technical aspect and capabilities but it also tempts them to show biased results. Consequently, the less transparent search engines likely to loss their credibility. Being a relative new market place, it has not much discussed in the academic and literacy spheres. This leded the search engine marketing to absence of operational definition of invalid clicks that is considered as major reason for lack of any adoptable framework against click fraud.

## References

1. Krol, C.: Search Engine Marketing Spending Surges. *B to B Magazine* **92**(3) (2007)
2. Best, R. In: *Market-Based Management: Strategies for growing customer value and profitability*. Volume 4. Pearson Prentice hall (2005)
3. Langville, A., Meyer, C. In: *Google's PageRank and Beyond: the science of search engine rankings*. Princeton University Press (2006)
4. Hedger, J.: The difference between paid and organic listings. <http://news.stepforth.com/2004-news/Jun02-04.html>, accessed 2007-04-22 (2004)
5. Fabos, B.: The commercial search engine industry and alternatives to the oligopoly. <http://eastbound.eu/journal/2006-1/contents/fabos/060109fabos.pdf>, accessed 2007-04-22 (2005)
6. Gerhart, S.: Do web search engines suppress controversy? [http://firstmonday.org/issues/issue9\\_1gerhart/index.html](http://firstmonday.org/issues/issue9_1gerhart/index.html), accessed 2007-03-14 (2003)
7. Eisenmann, T.R., Herman, K.: Google,inc. Technical Report 9-806-105, Harvard Business Review report, Harvard Business School, USA (2006)
8. Ellam, A.: Overture and google: Internet pay-per-click (ppc)advertising auctions. Technical Report LBS reference CS-03-022, London business school, London business school, UK (2003)
9. Wall, A.M.: Google adwords and yahoo ppc tips. <http://www.seobook.com/overture-adwords.pdf>, accessed 2007-04-29 (2006)
10. Marable, L.: False oracles: Consumer reaction to learning the truth about how search engines work. <http://www.consumerwebwatch.org/pdfs/false-oracles.pdf>, accessed 2007-04-20 (2003)
11. Jansen, B., Resnick, M.: Examining searcher perceptions of and interactions with sponsored results. Workshop on Sponsored Search Auctions at ACM Conference on Electronic Commerce (2005)
12. Nicholson, S.: How much of it is real? analysis of paid placement in web search engine results. *Journal of the American Society for Information Science and Technology* **57**(4) (2006) 448–461
13. Asdemir, K., Yaha, M.: Legal and strategic perspectives on click measurement. Technical report, School of Business,, University of Alberta, Canada (2006)



# Performance Evaluation of Slow Contention Window Schemes for Wireless Local Area Networks

Imran Siddique

Department of Computing Science  
Umeå University, Sweden  
int05ise@cs.umu.se

**Abstract.** IEEE 802.11 nowadays is a widely used wireless local area network standard. It uses Binary Exponential Backoff (BEB) to resolve the collisions between stations. In a shared medium a station selects a random backoff time period from a uniformly distributed Contention Window (CW). Upon each collision, a station doubles its CW value to reduce the risk of further collision, while on each successful transmission, the CW is reset. The resetting of CW causes the new collisions and then retransmission. The CW is more likely to decrease slowly. A simulation analysis of slow CW decrease and the BEB schemes are presented to determine the effective throughput, high fairness and less collision ratio. The simulation results show that slow CW decrease schemes significantly improve the performance of IEEE 802.11, as compared to the legacy of the IEEE 802.11 standard.

## 1 Introduction

Wireless networks nowadays are widely used and a great success after the deployment of the Internet. A wireless network allows users to connect to the network using waves instead of wires. There are two types of network used: centralized and distributed. A centralized network is centrally controlled which is called the *access point*. Distributed network has no central point. Every station accesses the network using a distributed function. IEEE 802.11 is a widely used standard to develop the wireless local area networks (WLANs).

In 1997, IEEE (Institute of Electrical and Electronics Engineers) launched the 802.11 standard for WLAN and also defined the Medium Access Control (MAC) protocol. The MAC basically controls the sharing of a transmission medium. MAC protocol defines two different access mechanisms, Distributed Coordination Function (DCF) and Point Coordination Function (PCF). PCF is a polling based technique which is centrally controlled and access is done through polling, where DCF is a multiple access technique based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism [1]. Using CSMA/CA, a station draws a random time from a time bounded interval, called Contention Window (CW). After the timeout and with no acknowledgement received from

destination, the sender realizes that the packet has collided. Upon each collision, the station increases the CW, to reduce the chance of further collisions. After each successful transmission, a station resets its CW to the minimum value of the CW size. Since it takes the minimum value of CW, the risk of the collisions is once again high. However, if we ensure that the CW decreases slowly that procedure outperforms the legacy of DCF in terms of throughput, fairness and minimizes the ratio of collisions[2, 3].

Several slow CW decrease schemes are introduced to overcome these problems as in [4], [5], [6], [7]. In order to study the performance of slow CW decrease schemes, two schemes, multiplicative and linear CW decrease, are evaluated. These schemes are easy to implement and give better results.

In the present paper my aim is to evaluate the performance of slow CW decrease schemes and compare with basic technique using the simulation results. Section 2, describes the mechanism of CSMA/CA in detail. In section 3, the three CW schemes, binary exponential backoff, multiplicative, and linear CW decrease are discussed. Section 4, is an evaluation part of these schemes with the help of simulation results and in section 5, I conclude this paper.

## 2 DCF (Distributed Coordination Function), A Basic Access Mechanism

DCF is a fundamental access mechanism for IEEE 802.11 networks which is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) technique. CSMA means that every station senses the medium to determine the current status of medium before transmission. If the medium is idle for a specific time interval, then the transmission is proceeds. Otherwise, the station waits until the completion of transmission [8]. When the medium becomes idle, a station performs a random backoff procedure, called Binary Exponential Backoff (BEB, see section 3.1), to reduce the probability of collisions with other stations that also wait for transmission.

In 802.11, all the packet types do not have the same priority [9]. So, the Inter Frame Space (IFS) time intervals have been defined, which gives a priority access control to the channel between transmissions [10]. There are two types of IFS are defined in the 802.11 protocol for DCF, DCF Inter Frame Space (DIFS) and Short IFS (SIFS). DIFS is the largest time period, comes prior to start of transmission, while SIFS is the shortest comes between data and acknowledgement (ACK) [1] as shown in figure 1.

The station senses the medium to check if it is idle for DIFS time period. If the medium is idle then the station starts transmission. After successful data transmission, the medium becomes idle again for the SIFS time period to prevent other stations from transmitting, while the receiver is sending ACK packet. The receiving of ACK at the sender side means that the data has been successfully received (see figure 1). If there is no ACK received in a specific time period, then the sender realizes that the data has been lost.

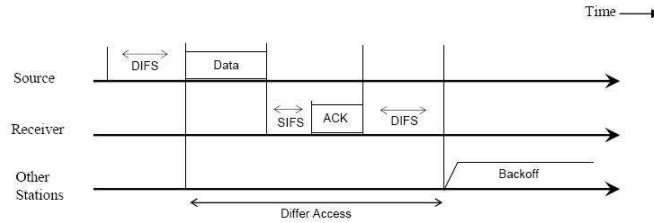


Fig. 1. A basic access mechanism of CSMA/CA (Redrawn from [1])

### 3 Contention Window (CW) Schemes

#### 3.1 Binary Exponential Backoff(BEB)

In the CSMA/CA, if two or more stations are trying to access the medium at the same time, then it leads to collision and the data is lost. To prevent this situation, the Binary Exponential Backoff (BEB) algorithm works along with CSMA/CA. After the DIFS time period every station counts down a random backoff value. One of the stations reaches backoff value at zero, it wins the medium and it starts transmission, while other stations pauses their backoff.

The random backoff value is drawn from a uniform distributed interval  $[0, CW]$ , where the CW is Contention Window. The initial value of the CW is set to be the minimum of CW size,  $CW_{min}$ . After each successful transmission a station resets its CW to the  $CW_{min}$ , while upon each collision, the collide stations double there CW, by using this formula  $(2x(CW+1) - 1)$ , until it reaches its maximum value,  $CW_{max}$ . Therefore it is called the binary exponential. The most commonly used values of  $CW_{min}$  and  $CW_{max}$  is 31 and 1023 respectively [1].

$$\begin{aligned}
 CW_{new} &= \text{Min} (2xCW_{prev}, CW_{max}) && \text{Collision} \\
 CW_{new} &= \text{Max} (CW_{min}) && \text{Success}
 \end{aligned}$$

Every station maintains its own CW interval and draws a random backoff value. After the DIFS time period, each station starts count down the backoff value until one the stations reaches zero and it starts transmission. Other stations pauses their backoff value and wait until the medium becomes idle again. After the successful transmission, the station resets its contention window to  $CW_{min}$ . Each backoff value is represented by one slot time.

The BEB, somehow reduces the chance of collision. But the collisions still occur if two or more stations reaches their backoff value at zero at the same time. This means that probability of collision is inversely proportional to CW size, i.e. a smaller CW size have greater probability of the collisions, while an

increase in CW size lowers the probability of collisions. But an increase in CW size creates more delay, low throughput and inefficient use of bandwidth [1].

### 3.2 Multiplicative CW Decrease Function

In the BEB algorithm, upon each collision the station doubles its contention window and on each successful transmission, a station resets its contention window to  $CW_{min}$ .

By taking the last point, i.e. a station resets its contention window to  $CW_{min}$ , taking again the risk of collisions. Aad et al. proposes [3] two slow CW decrease functions called, the Multiplicative CW decrease and Linear CW decrease for the DCF.

In the BEB, if two or more stations starts transmission at the same time as in the case of backoff reaches at zero at the same time, the stations waits until the timeout to realizes that the packets has been collided. The collided stations doubles their CW and retransmit the packet again. After the successful transmission a station resets its contention window to  $CW_{min}$ , it forgets the collision experience that it had. By keeping the history of collision, the multiplicative decrease function sets CW to 0.8 times its previous value, instead of resetting to  $CW_{min}$ .

$$\begin{aligned} CW_{new} &= \text{Min}(2x CW_{prev}, CW_{max}) && \text{Collision} \\ CW_{new} &= \text{Max}(0.8x CW_{prev}, CW_{min}) && \text{Success} \end{aligned}$$

By the slowly CW decrease, a small overhead included in the form of more backoff. But on the other hand it also saves the retransmission time. After few successful transmissions CW reaches again to  $CW_{min}$ . As compare to retransmission time, the overhead of slow CW decrease is negligible [3]. These both slow CW decrease schemes are easy to design and implement, as it is similar to DCF with just few modifications.

### 3.3 Linear CW Decrease Function

In the multiplicative decrease, it decreases by a multiple factor instead of resets CW to  $CW_{min}$ , as in the BEB. Similarly, the linear CW decrease function decreases the CW by a constant value called  $\alpha$ .

$$\begin{aligned} CW_{new} &= \text{Min}(2x CW_{prev}, CW_{max}) && \text{Collision} \\ CW_{new} &= \text{Max}((CW_{prev} - \alpha), CW_{min}) && \text{Success} \end{aligned}$$

In order to get good the results, the value of  $\alpha$  should be small, i.e. ( $\alpha < 100$ ). Aad et al. tests the different values of  $\alpha$  and suggests that  $\alpha = 50$ , for considerable throughput[3].



## 4 Performance evaluation and Discussion

### 4.1 Simulation Model

In order to investigate the slow CW schemes, I have design a model simulator in MATLAB. To reduce the complexity of simulation, several assumptions have been considered as in [11] :

- The channel data rate is assumed to be 2 Mbit/s.
- All nodes are stationary and in the range of each other.
- The effect of propagation delay are assumed to be negligible.
- The transmission is error free means that transmitted packet successfully and correctly received on the destination.
- Assuming the saturation condition, i.e. all senders have packets to send all the times. There is always a packet in queue.
- A collision happens only if two or more stations start transmission at the same time.
- There is no interference from the other nearby channels.

In the simulation, I assumed that stations operate at the IEEE 802.11b physical layer and their standard parameters are used. A list of parameters used in simulations are summarized in Table 1.

**Table 1.** Parameters used in the Simulation

Parameters	Values
Slot Time	20 $\mu$ seconds
SIFS	10 $\mu$ seconds
DIFS	30 $\mu$ seconds
Length of ACK	14 bytes
Data packet size	1460 bytes
CWmin	31
CWmax	1023
Data Rate	2 Mbps
Simulation Time	100, 2000, 3000, ... ,1000
Number of Stations	10, 20, 30, ... ,100

### 4.2 Evaluation

This section presents the performance evaluation of slow CW decreases schemes (Multiplicative and Linear) compared with the basic backoff scheme BEB, with the help of simple sceniors. The performance metrics that are used for comparison are as follows:

- *Throughput (Channel Utilization)*: The throughput is the rate of data transferring over the time that it takes and it is expressed in Mbits/s.
- *Jain's fairness Index* : To measure the fairness (equal opportunity to access the transmission medium) amongst the stations, Jain's fairness index [12] is used. Jain's fairness index is defined as

$$f(x_1, x_2, x_2, \dots, x_n) = \frac{\left(\sum_{i=1}^n x_i\right)^2}{n \sum_{i=1}^n x_i^2} \quad (9)$$

Where  $n$  is the number of stations and  $x_i$  denotes the measured throughput of station  $i$ .

- *Collision Ratio* : It is the ratio between the number of collision occurs over the number of stations.

#### 4.2.1 Throughput Comparison

**Description:** In this section, the throughput results of slow CW decrease schemes (Multiplicative and Linear) and the BEB are compared, for various number of stations. A simple scenario is considered with standard parameters and assumptions as describe in the above section. The simulation starts with 10 stations and runs for 1000 seconds. After every 1000 seconds 10 more stations have been added until the number of stations reaches 100. The average throughput is calculated for every 10 stations. The result is presented in figure 2.

**Results and Discussions:** As we can see in the figure, the slow CW decrease gains better throughput then the BEB, even with a high load, i.e. with up to 100 stations. As the number of stations increases, the throughput sharply decreases. The decrease is due to the increase in stations, which increases load of the medium. As a result more collision occurs and the throughput declines. There is less difference between slow CW decrease schemes and BEB over the 10 stations. It is because of less collisions at that time. It is observed that by using slow CW decrease, we gain a considerable throughput. That performance of slow CW decrease's throughput also presented in [3].

#### 4.2.2 Fairness Comparison

**Description:** In order to evaluate the fairness, Jain's fairness index  $F(j)$  is measured. To compute  $F(j)$ , the simulation runs up to 1000 seconds. In this scenario 100 stations are used throughout the simulation. The result is presented in figure 3.

**Results and Discussions:** The figure shows the fairness comparison between slow CW decrease schemes and the BEB. The BEB suffers the fairness problem, while slow CW decrease almost reaches to 1. The fairness is well known problem in BEB. As increases the number of stations that also increases the collision rate. As in the BEB, after every collision station doubles its CW, which reduces the chance of transmission, while other stations gets the chance to transmit their

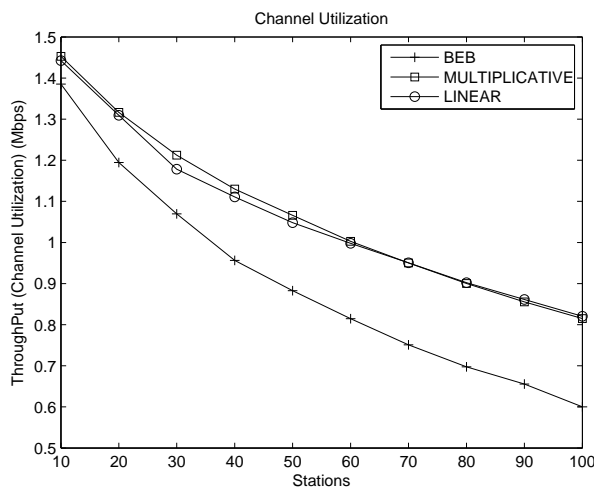


Fig. 2. Throughput Comparison between slow CW schemes and BEB.

data more frequently. That rises the fairness problem. However the slow CW decrease schemes fills the gap between CW<sub>min</sub> and high CW value, as result of collisions. The similar results are presented in [13].

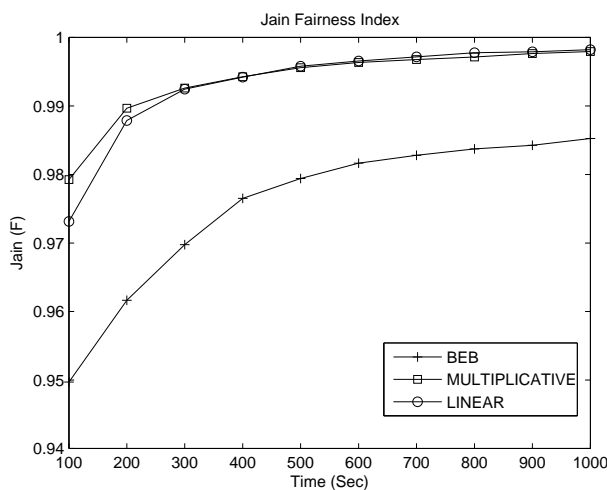


Fig. 3. Fairness Comparison between slow CW schemes and BEB.

### 4.2.3 Collision Ratio Comparison

**Description:** For the evaluation of efficient collision avoidance (CA) scheme, the ratio of collision is presented. The same scenario is considered as in section

4.2.1. The result is presented in the figure 4.

**Results and Discussions:** The figure clearly shows the difference between slow CW decrease and the BEB. The slow CW decrease scheme reduces the collisions all most 50%. The linear CW decrease scheme gives better results even than the multiplicative CW decrease. In the BEB scheme, reset the CW, increases the probability of collisions. This probability is less in the slow CW decrease scheme.

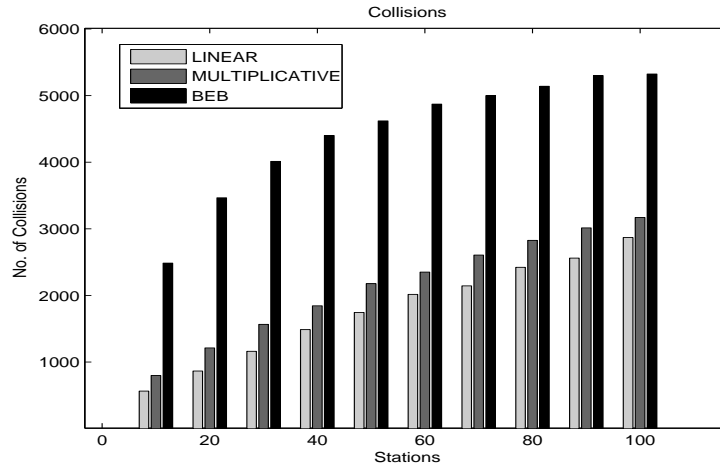


Fig. 4. Collision ratio of slow CW decrease Schemes and BEB.

## 5 Conclusion

This paper presents a comparative study of slow CW decrease schemes and the basic backoff mechanism deployed in the legacy of DCF. The DCF is based on CSMA/CA technique in which Binary Exponential Backoff (BEB) algorithm is works in order to avoid the collisions. A station senses the medium idle for a specific time period DIFS, before it starts the transmission. If the medium is busy it defer access. After DIFS time period, the station draws a random backoff value from a uniform distribution interval  $[0, CW]$ , where CW is Contention Window. Upon each collision a station doubles its CW size, while upon each successful transmission CW resets to CWmin. The stations forgot the collision experience by resets CW that it takes again the risk of collision. The BEB scheme suffers from fairness problem, inefficient channel utilization and delay.

To overcome these problem the slow CW decrease schemes are introduced. In the present paper, Multiplicative and Linear CW decrease schemes are evaluated. Instead of reset CW, the slow CW decrease scheme decreases CW slowly.

The simulation results are presented to show the comparison between slow CW decreases and the BEB. The results show that the slow CW decrease gains better throughput in both congested as well as non-congested environments.

It also achieves a high fairness under the congested environment. We can also see that the slow CW decrease schemes reduces the collision ratio all most 50%. Hence, the analysis shows that the slow CW decrease schemes reduce the collisions ratio, increase considerable throughput and provide fairness among the stations.

## References

1. Farooq, J., Rauf, B.: *Implementation and Evaluation of IEEE 802.11e Wireless LAN in GloMoSim*, Department of Computing Science, Umea University, Umea, Sweden (2006) 1–8
2. Pong, D., Moors, T.: *Fairness and Capacity Trade-off in IEEE 802.11 WLANs*, School of Electrical Engineering and Telecommunications, The University of New South Wales (2004)
3. Imad Aad, Qiang Ni, C.B., Turletti, T.: *Enhancing IEEE 802.11 MAC in congested environments*. In: Proceedings of Applications and Services in Wireless Networks (ASWN) IEEE, Boston (2004)
4. Bharghavan, V., A. Demers, S.S., Zhang, L.: MACAW: a media access protocol for wireless LAN's. Proceedings of the conference on Communications architectures, protocols and applications (1994) 212–225
5. J. Deng, P.V., Haas, Z.: A new backoff algorithm for the IEEE 802.11 distributed coordination function. Communication Networks and Distributed Systems Modeling and Simulation (CNDSS04) (2004)
6. Xiao, Y.: A simple and effective priority scheme for IEEE 802.11. Communications Letters, IEEE **7**(2) (2003) 70–72
7. Song, N., Kwak, B., Song, J., Miller, L.: Enhancement of IEEE 802.11 Distributed Coordination Function with Exponential Increase Exponential Decrease Backoff Algorithm. Proc. of VTC (Spring) (2003)
8. Zheng, L., Dadej, A., Gordon, S.: Fairness of IEEE 802.11 Distributed Coordination Function for Multimedia Applications. Proceedings of the 7th International Conference on DSP and Communications and 2nd Workshop on the Internet, Telecommunications and Signal Processing, Coolangatta, Australia (2003) 404–409
9. Aad, I., Castelluccia, C., INRIA, R.: Differentiation mechanisms for IEEE 802.11. INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE **1** (2001)
10. Kuo, W., Kuo, C.: Enhanced backoff scheme in CSMA/CA for IEEE 802.11. 58th IEEE Vehicular Technology Conference **5** (2003)
11. Natkaniec, M., Pach, A.: An analysis of the backoff mechanism used in IEEE 802.11 networks. Computers and Communications, 2000. Proceedings. ISCC 2000. Fifth IEEE Symposium on (2000) 444–449
12. R. Jain, D.C., Hawe, W.: A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. Technical Report TR-301, DEC Research Report (1984)
13. Aad, I., Ni, Q., Barakat, C., Turletti, T.: Enhancing IEEE 802.11 MAC in congested environments. Computer Communications **28**(14) (2005) 1605–1617



# Frameworks for Context Aware Ad Hoc Communication Systems—A Survey

Khurram Ali Khan

Department of Computing Science  
UmeåUniversity, Sweden  
ens03kkn@cs.umu.se

**Abstract.** Recent research in context aware computing emphasises mainly the development of common architectural framework for contextual data acquisition. The existing frameworks in this domain highly depend on problem specification, which leads to different architectural design approaches. This paper will provide an overview of three different approaches for a common framework architecture in context aware ad hoc communication systems. The intentions are, not to conduct extensive comparison among these architectural models but to present a study on their context models, implementation tools and also discuss some features they provide in their particular problem domain. The work can be used to compare and study these models.

## 1 Introduction

The introduction of context in computing expands the semantic domain of computing systems. Their collaboration with mobile devices in this scenario brings new and vast opportunities to developers as well as to end users. The term “context” in computing environments is defined by many researchers according to their needs. The most referred definition by researchers now, is given by Dey and Abowd [1–3]. According to Dey et al. [4], “[c]ontext is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and application themselves.”

According to Dey et al. [4], systems that can alter their functional behaviour based on the context of the application and the user environment, can be termed as context-aware systems. They provide the following definition [4]: “A system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user’s task.”

A context aware computing systems is a collection of handheld devices, wearable computers, sensor systems, internet or network connections. Because of the high vulnerability of such devices in real world environments it is necessary to make their use invisible [5].

Currently, the available context aware systems depending on mobile devices and their physical environments are usually built considering specific purposes. They are usually not extendible and unable to adapt to even the slightest change

in their problem domain [6]. Baldauf et al. [1] mention that the development of context aware systems depends on the number of users and the devices used in data collection and their location. In context aware systems the method of getting contextual data is vital as it defines the architecture of the system [1].

Winograd et al. [7], recognize that different architectural approaches have been taken to produce common frameworks for context data gathering, but so far there are very few. These frameworks can be distinguished according to their architecture, context model, implementation approach and the specific features they cover in context sensitive ad hoc communication systems [7].

Three types of architectural framework models are discussed in this paper. The remainder of this paper is organized in this manner: Section 2 describes the Middleware architectural approach by presenting two existing models. Section 3 will present a blackboard based approach by discussing context management frame work developed by research members of VTT technical research centre of Finland. In Section 4, the Hydrogen project is presented which is a three layer architecture. Section 5 will discuss some of the features and present a summary in the form of table. Section 6 concludes the paper.

## 2 Middleware architectural approach

Middleware is any software/application that is used to connect different parts of applications/software in heterogeneous or homogeneous environments. Middleware architectural approach produce a layered architecture for collecting contextual data in context aware computing. This architectural approach is usually used in context sensitive ad hoc communication systems. They are the systems that acquire context data from various local or remote mobile sources and use communication links among them to communicate and for data exchange.

There are many different implementations of middleware approaches for context aware systems. This section will present two of them; reconfigurable context-sensitive middleware and Cortex.

### 2.1 Reconfigurable context-sensitive middleware

Reconfigurable context-sensitive middleware (RCSM) is a middleware context data acquisition approach, that is used by the systems that are context aware and require ad hoc communication among devices or application. It is developed by the research team of Arizona State University [5].

RCSM is an object based development framework. It is placed in between context aware application and ad hoc communication as in (figure 1), thus making a middle layer. According to Yau et al. [5], RCSM lets application software directly communicate with each other and provide message oriented semantic between object/devices through remote procedure.

In RCSM the context sensitive application is modelled in context-sensitive objects, these objects have two components, a context sensitive interface, which



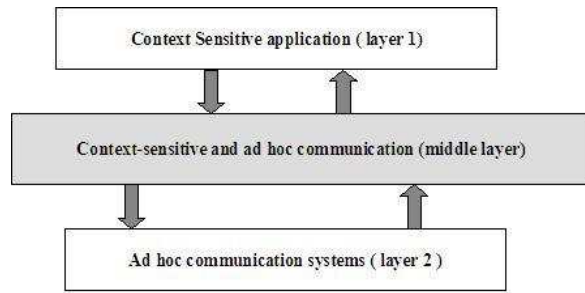


Fig. 1. The middle layer is RCSM, layer 1 is the context sensitive application and layer 2 is the ad hoc communication system [5].

contains the information related to the contextual data of the application while the other contains the implementation details [5].

In RCSM data is treated as event because most of the data comes from different sensors and mobile devices. To describe context's event and methods RCSM used "context-aware interface definition language" (CA-IDL), it is a compiler to generate object called 'Adaptive object container' these are context sensitive object and use to communicate with implementation component to activate action with respect to the current context [5].

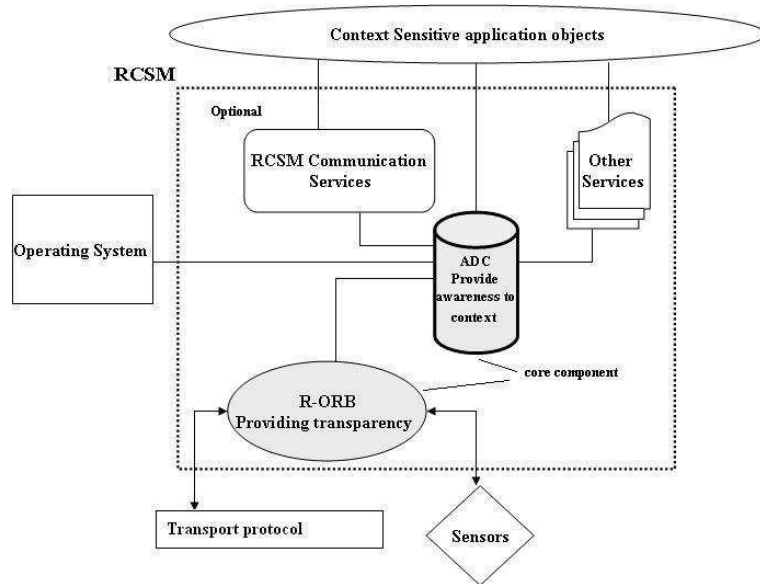


Fig. 2. RCSM structural model, adapted from [5].

RCSM object request broker (R-ORB) is a mechanism used to make communication transparent between different computational devices and sensors used in applications. Service and device discovery is also performed by R-ORB according to the appropriate context specification for the particular object. The communication link between two remote devices managed by context triggered point-to-point mechanism, this link is established and maintained on RCSM general inter object request broker protocol (R-GIOP) [8]. Different parts of RCSM are shown in (figure 2). The structural model consist of two main components core and optional, the dotted lined box shows the RCSM model between context sensitive application and ad hoc communication system.

**Features.** The framework of RCSM is object based, its implementation is separate from the contextual data which means that the developer need not to worry about context monitoring, detection and analysis, this property makes it useful for those systems which require specific action when multiple sensory and mobile devices produce certain data to trigger an action. The architecture is useful for location aware systems and for those, which require continuous context monitoring. Symmetric communication is used by this model. The use of object-oriented model provides encapsulation, reusability and inheritance. The authors of this system, presents only one example where it is used, the “smart classroom” [8].

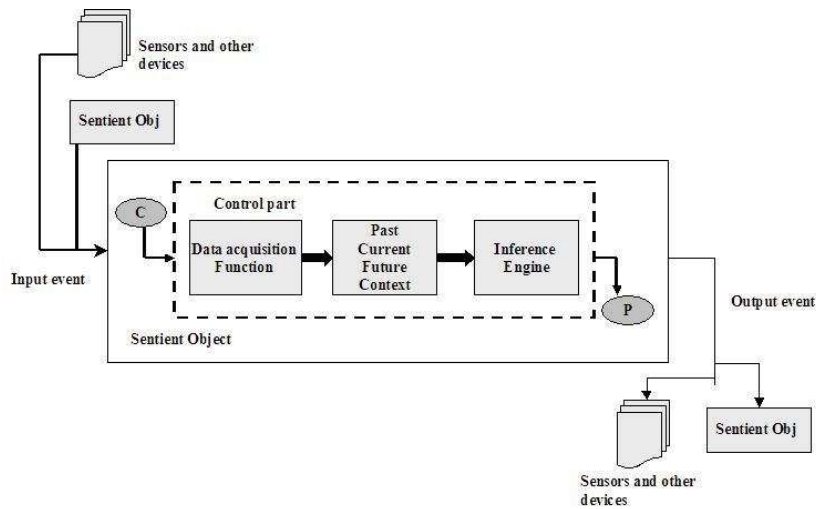
## 2.2 Cortex

Cortex is another implementation of the middleware architectural approach towards the acquisition of context data in context-sensitive and ad hoc communication environment [1].

Cortex is based on sentient object Model. The object model consists of sentient object which consist of three parts, context data capturing, context hierarchy and an inference engine, figure 3 shows the sentient object and its parts.

The sentient object receives context data as input events from different sensors via its interface, then based on past, current and future context data it is processed and then analysed by the interface engine to acquire high-level context data, that is produce as input to other application devices or sentient object in the form of event. The mechanism is useful for getting high-level context data from low-level [9]. The important aspect of the sentient object is that it can contain number of sentient object in itself that makes them both producer and consumer of events.

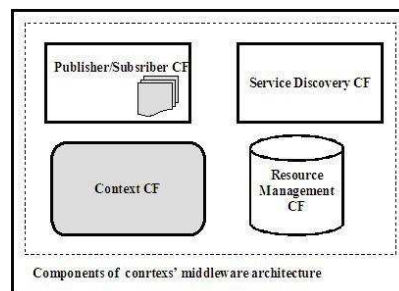
The cortex middleware architecture consists of: publisher/subscriber, service discovery, reason management and context component frameworks as shown in figure 4. The component framework can be defined as a collection of rules and interfaces that used in the interaction of component. The publisher pushes events into event model and the subscriber receives event, XML is used for the representation of event model and the communication of event across network is achieved by XML based protocol named SOAP. In context aware mobile systems various types of events are published and subscribed, which might raise the possibility



**Fig. 3.** Sentient object model [9]. C and P are consumer and producer sentient object, the control part of the object contains capturing function, stored and current data and some inference mechanism.

that subscriber are unaware of their present context send by different publishers. To over come this problem service discovery component framework is used, it has its own interface for the services it provides. The two implementation of service discovery protocol that Cortex is using are SLP and UPnP. Resource management framework is responsible for managing resources, these resources can be system resources. It is useful in prioritising the event produce by publisher [9].

**Features.** The implementation of cortex is done in OpenCom developed by Lancaster University using features of COM given by Microsoft. Cortex has a relational data model. This model best suits computing environments where connection between different resources cannot be predicted or are not regularly



**Fig. 4.** Cortex middleware components [9].

available. Event sending mechanism in this model, send events to only those devices/application that are subscribed for that particular event, hence reducing the network traffic, context sensitive ad hoc communication systems usually have many devices and generate different events for particular context, the model can prioritise those events [9].

The system needs the development of the context component framework, as the authors [9] provide no explanation and use of this component in his work. The system is in implementation face at the time of citing, only one example scenario is given by the authors, according to authors [9], application level multicast is also being developed and more features will introduce related to component framework in collaboration with University of Lisbon [9].

### 3 Context Management Framework

The Context Management Framework is built by the members of VTT Technical Research Center of Finland, The framework is built for those systems that contain mobile devices, sensors and ad hoc networks in their computing domain, its architecture is based on blackboard model approach. Blackboard can be defined as an entity that can be access and shared by different process or application exist in particular computing environment. The Framework architecture consists of four components as shown in figure 5, the components are: context manager, resource server, context recognition service and application component [10] [1].

Different sensors systems and mobile devices that are used by the system to get context data are connected by the resource server. There are four process of resource server:

- Sensor management: gets the raw data from the connected sources.
- Preprocessing: compare the raw data with the sample data which is explicitly entered.
- Feature Extraction: calculating more specific context from the raw data.
- Quantization and Semantic labelling: this process of resource server give meaning to data by combining feature value to the physical environment [10].

The final output data from resource server is sent to context recognition services, this component contains a table of recognized services, its work is to identify the data and its context, represent it in the form of 'context based ontology vocabulary' for context manager [10].

Context manager component work as a black board in this architecture (see figure 5), each device or application communicates with context manger, in the form of client-server relation. Services that context manager provides, can be local or distributed. Context manager store information of contextual data and all the terminals or application attached to it can get access to the context data by querying it or by subscribing for various events [10].

**Features.** The system can be extendible because of the use of ontology which lets the developer add different features. Those mobile devices that support

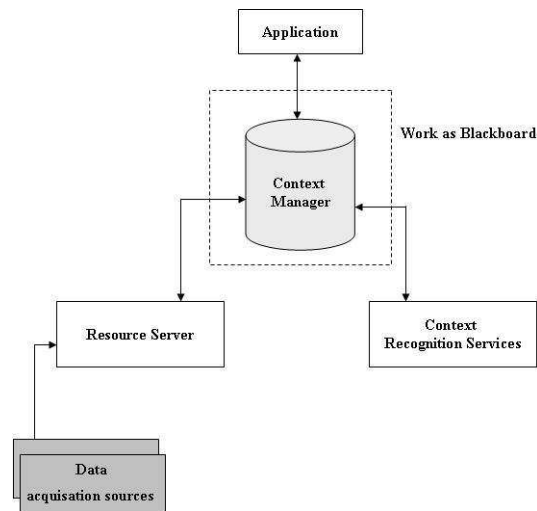


Fig. 5. Components of Context Management Framework [10].

GSM, GPRS, Bluetooth and ad hoc networks, can be used as data collecting sources. Context Management Framework uses ontology for context representation and for sharing information they use 'Resource description framework'. Context manager API is implemented on Symbian platform, for acquisition of high level data from low level, Bayesian algorithm is used [10].

The framework requires explicit data entry for comparing different levels of contexts' data abstraction which requires a large amount of data. According to Kela et al. [10], Application and context manager exist on mobile devices which implies that they should contain high computing capability. Sampling data that the system uses, is not a real-time data which may cause problems while implementing in real-life scenarios [10, 1].

## 4 Hydrogen Approach

This three-layered architecture is suitable for context-aware mobile device systems. The system is built by research members of Software Competence Center Hagenberg, Austria. In mobile devices like PDA, cellular phone and Blackberry, context awareness is complex because the context is highly dynamic and mobile and furthermore these devices do not have much computing power, such characteristics of systems generate restrictions in context-data acquisition framework architecture [11].

Hydrogen approach is a three-layered architecture consisting of adaptive layer, management layer and application layer (see figure 6), which are located on the same device.

The physical context data is gathered by the adaptive layer and this data is usually generated by different mobile devices. The contextual information

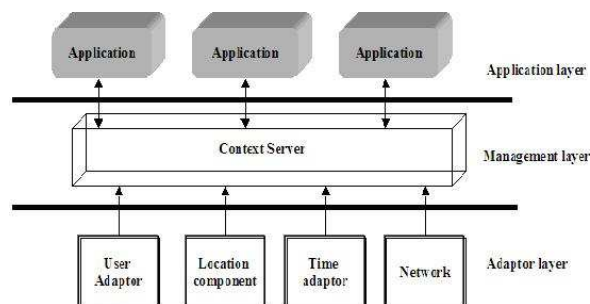


Fig. 6. Three-layered architectural approach of Hydrogen project [11].

is forwarded to context server that positioned in management layer, the work of management layer is to produce and receive data among devices and application. It uses peer-to-peer communication for this purpose. The communication can be asynchronous in which application can query data from context server and in synchronize communication application is informed by the server about the changes occur in context. Application layer contains all the applications for particular system [11].

According to Hofer et al. [11] In Hydrogen approach the context is divided into remote and local context data, the contextual data that one device contain of another is called remote while the local context data is devices' own data. This three-layered architectural frame work avoid the dependency of centralize component which most existing context aware systems contain [11].

**Features.** The implementation of Hydrogen project is based on J2ME, Java virtual machine and J9 [11].The communication is done by using XML based protocol, most exchange of data is done using XML-stream as the data resides in different architectural devices, thus given interoperability and compatibility, the context architecture is done by using UMLs' class diagram. The system is most useful for application where mobile devices are used, it is light weighted system, the architecture support connection to remote sensors, two or more application can access the same contextual data simultaneously as the data gathering, storing and its use is separate. Discontinuity of network connection is over come by locating three layers on one device [11].

## 5 Discussion

Four existing models Reconfigurable context sensitive middleware, Cortex, Hydrogen and Context management framework presented in this paper, although there are others like CoBra, Context toolkit and many more but they use different approaches for example, widget based and agent based, which seems to be out of scope of this paper.

RSCM and Cortex uses middleware architectural approach but they differ in context model, because of the use of object oriented methodologies RSCM can be modified where as cortex has limitation when it comes to integration. The authors of both these system implement them in similar environment like smart class room and intelligent room, which gives an idea that these frameworks still emphasising specific domain. The implementation of these two frameworks based on Java and XML. Hydrogen uses three layered architectural approach, the framework focus on an important issue of context aware communication systems that is , the discontinuity of network. It also use object oriented paradigm towards it context model. If you consider middleware or three layered architectural approach it seems that security is no where near an issue for the authors of these system although its has been mentioned that context aware communication systems are highly vulnerable to physical environment. Context management framework is a blackboard approach based on the technology of W3C Resource description framework. In this system the component that work as blackboard is context manager and if it fails or unable to work accordingly the whole system will be in jeopardy. This is also a concern issue when it comes to frameworks that based on middle ware. Table 1 gives the summary of discussed framework architecture and their features.

**Table 1.** Summary of discussed frameworks architecture.

Framework	Architecture	Context model	Tools	Implementation	Security
RSCM	Middleware	Object Oriented	Java, XML, C++, CA-IDL, R-ORB.	Smart class-room	n.a
Cortex	Middleware	Relational data model	XML, Open-Com, SOAP, UPnP.	Intelligent room,co-operating car application	n.a
Context Management Framework	blackboard based	Ontologies, W3C based Resource description framework	Ontologies based API, Symbian, XML.	Sensor-based context information in mobile application	n.a
Hydrogen	Three layered-architecture	Object Oriented	J2ME, XML, TCP/IP, JVM, UML.	Context aware post box	n.a

## 6 Conclusions

The paper discussed four existing models for common architectural framework aimed at contextual data acquisition in context-sensitive ad hoc communication. It also presents different features related to these models, the focus is mainly on

implementation tools and framework features. The four models use three different architectural approaches. Each model has their own benefits, although the authors of these model implemented and tested them in. Context-sensitive ad hoc communication systems contain different mobile and sensory system, data is usually unstructured and communication between different raw data is difficult. Considering this, most of these model use XML and XML-based protocol. Although the data in context-sensitive ad hoc communication is vulnerable , the systems presented are inadequate when it comes to security issues. The future work in this area will be to unleash other architectural approaches and in-depth analysis of their working models, also to study the environments in which these systems will implemented.

## References

1. Baldauf, M., Dustdar, S., Rosenberg, F.: A survey on context-aware systems. *International Journal of Ad hoc and ubiquitous computing* (2004)
2. Moran, T.P., Dourish, P.: Introduction to this special issue on context-aware computing. *Human Computer Interaction*, Volume 16, pages 87-95 (2001)
3. Przybilski, M., Nurmi, P., Floreen, P.: A framework for context reasoning systems. *Proceedings of the 23rd IASTED International Conference on Software Engineering*, Pages 448-452 (2005)
4. Dey, A.K., Abowd, G.D.: Towards better understanding of context and context-awareness. In *Workshop on the What, Who, Where, When, Why and How of Context-Awareness at Conference on Human Factors in Computing Systems* (2000)
5. S.Yau, S., Karim, F., Wang, Y., Wang, B., K.S.Gupta, S.: Reconfigurable context-sensitive middleware for pervasive computing. *IEEE International Conference on Pervasive Computing*, Volume 1, pages 33-40 (2002)
6. Przybilski, M., Nurmi, P.: An architecture to enable remote context reasoning. <http://www.cs.helsinki.fi/u/ptnurmi/papers/PSC3042.pdf>, last visited 2006-05-02 (2006)
7. Winograd, T.: Architectures for context. *Human Computer Interaction (HCI)*, Volume 16, Pages 401-409 (2001)
8. Buchmann, D.: Reconfigurable context-sensitive middleware for pervasive computing. <http://homepage.hispeed.ch/budda/downloads/rcsm.pdf>, last visited 2006-04-25 (2006)
9. Hector, A., Limon, D., Blair, G.S., Friday, A., Grace, P., Samartzidis, G., Sivharan, T., Maomao, W.: Context-aware middleware for pervasive and ad hoc environments. <http://www.comp.lancs.ac.uk/computing/research/mpg/projects/cortex/archive/dmrg/20publications/cortexMiddleware.pdf>, last visited 2006-04-26 (2006)
10. Korpipää, P., Mäntyjärvi, J., Kela, J., Keränen, H., Malm, E.J.: Managing context information in mobile devices. *IEEE Pervasive Computing* **2** (2003) 42-51
11. Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., J, J.A., Retschitgegger, W.: Context-awareness on mobile devices-the hydrogen approach. *System Sciences, proceeding of 36th annual Hawaii International Conference* (2003)



# A multilayered decision model for command and control systems

Mikolaj Kunc

Department of Computing Science  
Umeå University, Sweden  
ens03mkc@cs.umu.se

**Abstract.** This paper presents a multilayered approach towards the creation of Command and Control (C2) decision models. The Multilayered Decision Model provides a proper level of the description of the decision making process in all echelons. It takes into account the different types of complexity required at the different echelons and shows how the information propagates throughout the whole command structure. It can be used as a basis for a C2 system or as a tool to model communication between the echelons. The multilayered decision model offers structure that can help to solve the time delays problem identified by Brehmer [1] and offers possibilities of further development towards creating a system that will be able to describe phenomena appearing not only on one command level but also between levels in the whole command structure.

## 1 Introduction

Command and Control (C2) systems play an important role in the modern military forces. They provide tools to support the decision making process and minimize the time required for the commanders to make proper use of the forces that are available to them. As C2 we should consider all control activities that are preformed by military at all time. C2 gives the activities a meaning and synchronizes them together so that the actions that are preformed at various echelons are coordinated and serve one common goal. We will use the definition of the US Marine Corps “Command and control is the means by which a commander recognizes what needs to be done and sees to it that appropriate actions are taken” [2]. Starting from this general definition we can now consider what is necessary to design a well working C2 system. Defining a functional specification is essential as Brehmer [1] points out if the design process starts from creating the physical form of the system then it is very likely that, given the purpose, the outcome of the design process might be far from expected. Knowing the functions the next step would be to create the system. If we take into account the cyclical nature of all military activities (incoming reports and outgoing orders) the most obvious structure for a command and control system would be a loop. The loop should describe the data flow and be the actual skeleton of the whole command and control system. The first loop model was called the Observe-Orient-Decide-Act (OODA) decision model [3]. Since then the models have been modified to face the evolution of the modern battlefield.

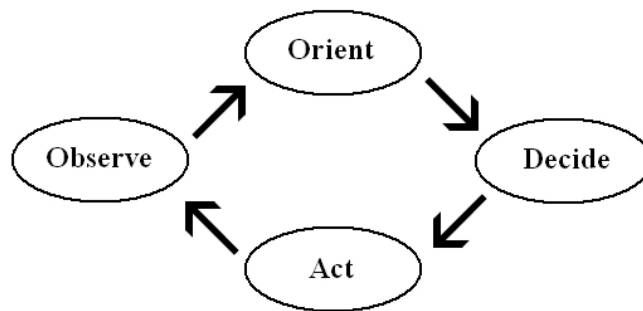
This paper questions the traditional approach towards the command and control systems. I will try to show that by using a multilayered C2 structure we can achieve a

higher level of control over our own forces. I will show that single loop decision systems have problems with providing proper levels of description for an echelon where they are used. They are either too complex to use or do not have enough connection to the environment.

In this paper chapters 1–2 introduce a range of methods that are currently used in the development of C2 systems and analyze their utility in the multilayered model. In chapter 3, I propose a solution to the problems described in the previous parts of the article. Chapter 4 is a discussion of the solution's advantages and disadvantages.

## 2 Analysis of the existing decision loops

In the first step I will analyze the existing loops and see how they manage to solve problems of multilayered decision system. The analysis will cover both the original OODA loop and the two modifications: COODA and DOODA loops.



**Fig. 1.** The OODA loop (redrawn from [4])

It is worth mentioning that the first control loops that were used to improve decision making process and quality were created in the first part of the 20<sup>th</sup> century. In 1939 Walter Shewhart suggested a model [5] that should provide continuous improvement for the process for which it is run. The loop is called Shewhart cycle and consists of four elements: Plan, Do, Check, Act (PDCA), which in name and function are similar to the elements that the OODA loop is composed of. In modern times decision support systems (DSS) play important role in many aspects of our lives (e.g. medical diagnosis, logistics, business and management). With progress in the field of artificial intelligence and increased computational power it might be possible that the quality of their decisions will remove “support” from their name [6].

### 2.1 The OODA loop

The Observe-Orient-Decide-Act loop (OODA) was developed by John Boyd [3] to explain the air superiority that was achieved by the American pilots over their adversaries

during the Korean war. As can be seen on figure 1 the loop consists of four elements that create a coherent structure that was (and still is) used as a basis for C2 decision systems.

The phases of the loop are described below:

- Observe - During first stage of the loop we sense the environment and gather information about it. In military terms it means “gathering information about own forces, the enemy, the weather and the terrain” [1].
- Orient - Second stage will asses the situation, create objectives and ways of achieving them.
- Decide - From all the possible courses of actions that can lead to a desired goal one is chosen. Orders are written and transmitted.
- Act - Implementation of actions according to received orders.

Each phase of the loop produces an output that is passed to the next one in the cycle. This representation allows us to identify at least one important aspect of C2—the time constraints [7]. One of the purposes of this system should be to minimize the delays between the phases. This model also has disadvantages. It is too abstract and has in fact no connection to the environment. Brenton and Rousseau in their paper [7] identify this problem as a low level of cognitive granularity. On a higher level of command this is not acceptable. Nevertheless the unpredictability of the environment makes this model a good basis for a C2 for lower echelons. A problem related to this one is that the user might experience difficulties in creating a clear definition of what kind of input information that should be provided for each state and also to what kind of output information that should be found on the outputs.

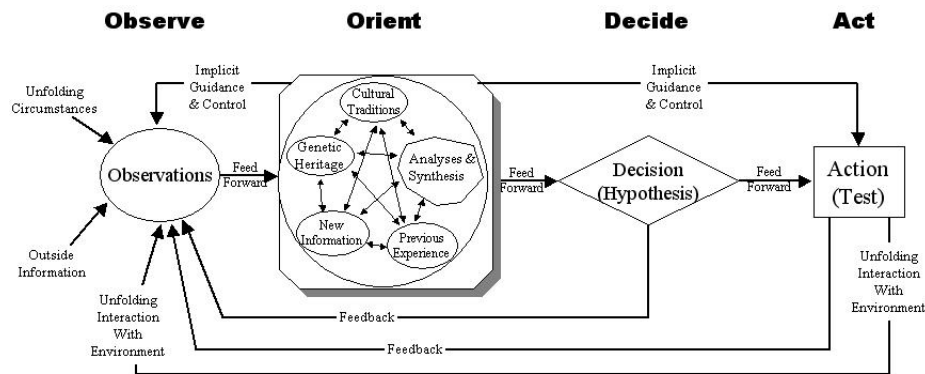


Fig. 2. The Extended OODA loop (courtesy of Chet Richards(www.d-n-i.net))

The Extended OODA loop (figure 2) introduced by Fadok, Boyd and Warden [8] tries to solve the problems described above by adding feedback and feed-forward loops and increasing the complexity of the Orientation stage. The new OODA loop is centered

around the complex version of the Orientation stage which influences all other elements of the loop. It is also worth noticing that the inputs to the Observation stage were described with a higher level of detail. This improvement together with the increased level of cognitive granularity in the Orientation process better supports connecting the loop to the environment. Unfortunately the choice of factors in the Orientation stage that are supposed to influence the final decision is not realistic. Commanders usually have too little time to consider e.g. “Genetic Heritage” in the decision making process.

## 2.2 The Cognitive OODA loop

A modern version of the OODA loop was proposed by Robert Rousseau and Richard Brenton [7]. In the Cognitive- OODA loop (figure 3) the three first phases (Observe, Orient and Decide) have been extended with a set of analysis and control functions that can alter the data flow. For example the Observe stage has Perceiving and Features Matching phases responsible for analyzing the incoming data whereas Data Clear and Familiar control the processing sequence. A new data flow has been introduced. Feedback functions are implemented between neighbouring phases which makes the data flow bidirectional. The only exception is lack of feedback between Observe and Act phases. This version of the OODA loop is designed with a high level of cognitive granularity and is able to produce a valuable decision in a harsh, time restrained environment.

Knowing that the loop will be used as a C2 model it is worthwhile asking who will be using it. Its complexity and in some cases the need for intuition and experience enforces that at least some of the operations in this loop will have to be performed by people. The following question should also be asked: at what level of command should this loop be used? It cannot be used at lower levels (e.g. platoon, squad) because it's structure is too complicated and decisions on that level of command should be done as quickly as possible. Leaders here should rely more on their experience and intuition than on calculations and evaluations. Another problem is that on a higher echelon (e.g. company) the initial planning loop (after receiving the order) will be different from the ones that will occur after the first Act phase takes place. The initial planning will take more time and preparations whereas in the following iterations the Orient and Decide phases will be much shorter and will only have to adjust the initial plan to the situation on the battlefield. This has not been taken into account.

The division of the OOD stages to analysis and control functions makes it difficult to use this loop in lower echelons (because the functions are too complex and time consuming). However, in upper echelons they would help to maintain the time discipline and high quality. Therefore this loop is a very good candidate for a C2 decision model at a higher level of command.

## 2.3 The Dynamic OODA loop

Our last loop is the Dynamic-OODA loop (figure 4) proposed by Brehmer [1], which is intended to be a general model of C2. The designers wanted to create a decision loop that describes which functions that should be involved in the command and control

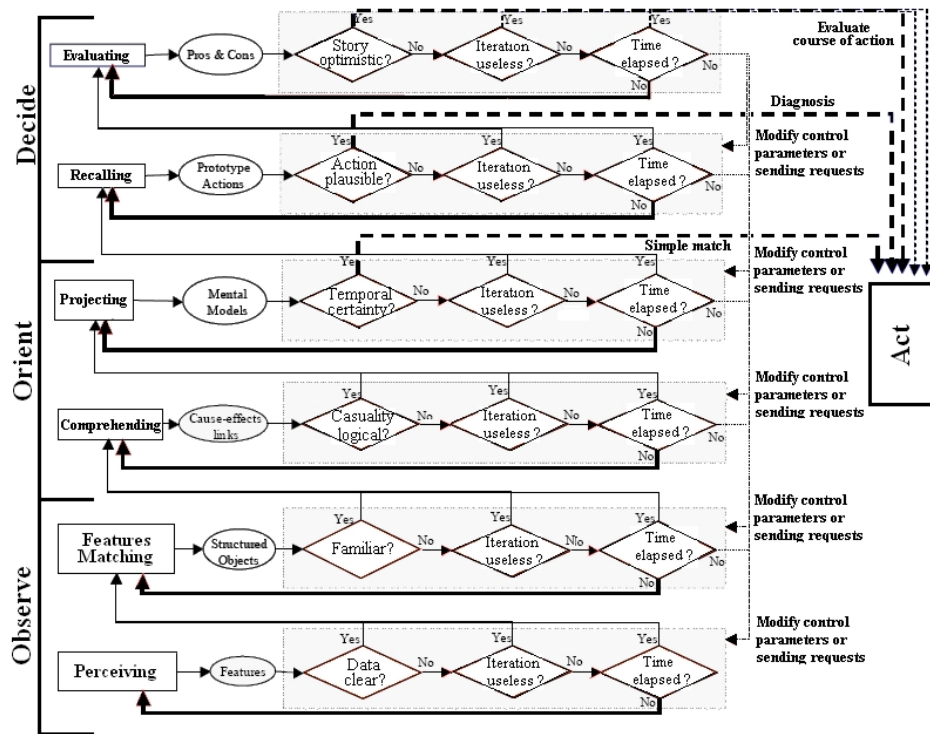
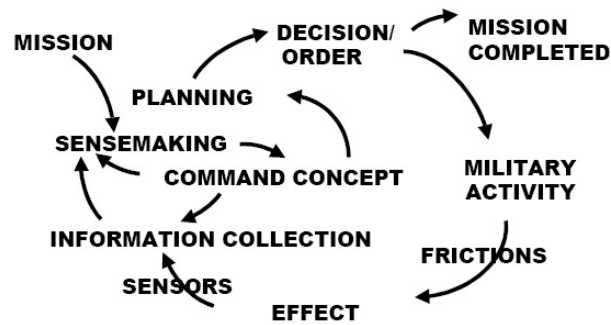


Fig. 3. The Cognitive-OODA loop (courtesy of Richard Brenton [7])

system. To do that they used the eight functions defined by van Creveld [9], which are described below:

1. Gathering information on own forces, the enemy, the weather and the terrain
2. Find means to store, retrieve, filter, classify, distribute and display the information
3. Assessing the situation
4. Laying down objectives and working out alternative means for attaining them
5. Deciding what to do
6. Planning
7. Writing orders and transmitting them as well as verifying their arrival and proper understanding by the recipients
8. Monitoring the execution by means of feedback at which the process repeats itself.

What distinguishes this structure from the others is the start and end points for the loop (the mission and mission completed stages). These elements are very important if we consider the loop as a part of a more complex, multilayered system. The DOODA loop is a good example of high level decision loop. The planning part is well described and the Act part is minimized.



**Fig. 4.** The Dynamic-OODA loop(courtesy of Berndt Brehmer [1])

#### 2.4 Loop summary

The main problem that can be found in all the models above is that because only one type of loop was developed it is obvious that this loop in each cycle has to produce a decision. Attempts have been made to make sure that the decision will be as accurate and correct as possible (i.e. using the time to the maximum by introducing the “Time elapsed” control function in the COODA model). Nevertheless if we consider for a moment a model of commanding where the higher level of command gives more freedom and opportunities to for initiative to the lower echelons then we can come to the conclusion that it is not necessary to give a very detailed instructions. The goal of the loop would be to provide the commanders of the lower echelons with the overall picture, the intentions of the superior officer and what is expected from them. It is clear now that different levels of command will focus on different aspects of C2 and therefore should use different types of decision loops. The complete command and control model in this case would be a layered structure with each layer representing loops for different echelons. The natural connection between levels would be the Act phase where the orders will be transferred to a lower level and the reports will be received at a higher level of command.










### 3 Multilayered Decision Model

Taking into account conclusions drawn form the analysis of the previous OODA loops we can establish what type of characteristics that the C2 model should have. I would suggest that apart from the functions described by van Creveld [9], we should also consider the following factors:

- Different echelons take into account different aspects while making a decision
- The level of details in orders should increase while going down the command structure (down the echelons)

- At the lower echelons the loops run much faster than in upper, therefore they should be simpler
- The first iteration usually is more focused on planning than on the action itself
- The Act phase takes place ONLY at echelon levels where there is a contact with the enemy
- To encourage the initiative and reduce the necessary amount of control over own forces what is passed in orders (until platoon level) is commanders' intentions and not the detailed instructions [10]

**Table 1.** Symbols and abbreviations

Symbol	Explanation
O	Order
R	Report
MC	Mission Concept
INT	Intelligence
P	Plan
BD	Battle Drill
METT-TCSL	Mission Enemy Terrain Troops - Time Civilians Space Logistics Activity
	Object passed between activities
	
	Communication with a lower echelon
	
	Regiment
	Battalion
	Company
	Platoon
	Squad

The general rules that are described above apply to all layers of the model, but apart from them each layer has its own set of characteristics that are unique. This criteria supports the theory that one, general decision loop cannot be used to represent C2 for all echelons.

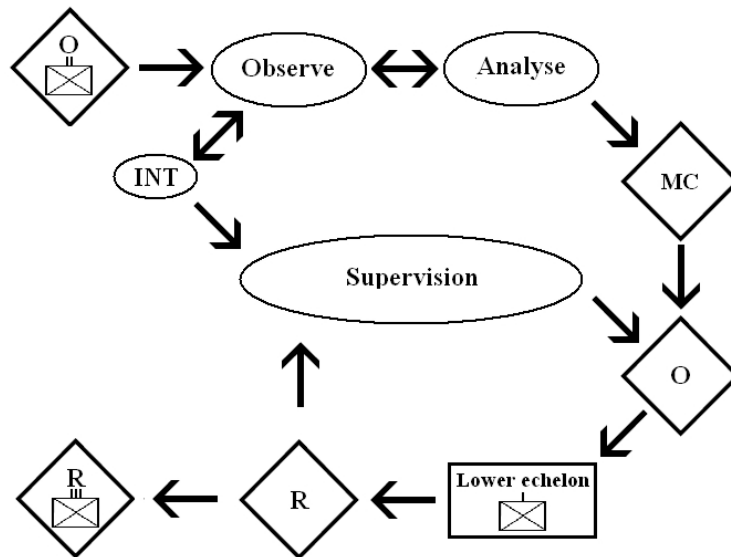
In the following solution I will present loops only for four echelons: battalion, company, platoon and squad. Capt. Bazin [4] shows that even a single soldier uses a simple

OODA loop during performing his tasks. All abbreviations and symbols used in the figures are explained in table 1.

**3.1 Battalion (4 companies, 640 personnel)**

A battalion is a military unit usually consisting of from 300 to 1000 personnel. In this case I assume that this unit consists of 4 companies, consisting of 160 personnel each. It is obvious that the commander of such a unit is not able to issue orders for each and every soldier. If we consider the time limits and area that is being controlled by his battalion, he is not even able to create very detailed orders for his company commanders. Therefore he creates a concept of a mission [10]. This structure defines the mission task and the result but does not describe in detail how the mission should be accomplished. This saves the time on the battalion level and encourages the initiative and creative planning at lower echelons.

The second issue here is that due to the fact that the orders that are created in this



**Fig. 5. The Battalion Control Loop**

loop are more like general “guidelines” than a detailed list of actions to perform, the control of following the orders should take the form of a supervision rather than control. In that case a full OODA loop is not necessary and the new orders will just be additional guidelines and not completely new concepts.

Based on these assumptions the Battalion Control Loop (BCL) might look like on the figure 5. The loop starts with the battalion commander getting an order (a mission



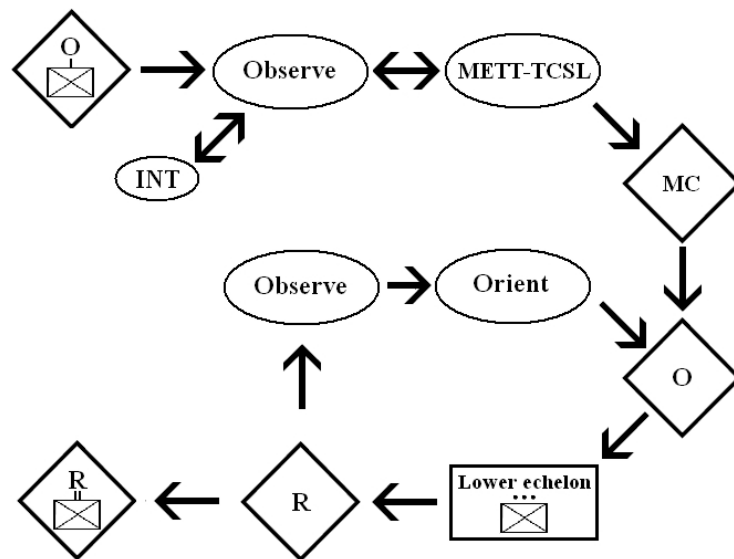
concept) from his superior officer (e.g. a regiment or a brigade commander). The Observation stage is fed with the mission concept and the data provided by the intelligence. It refines the data and produces information that can be used in the Analysis of the situation and in the Planning of the solutions for his company commanders. The MC is created and sent to units in the form of an order.

The supervision part of this structure starts after the units receive their orders and it may take a form of providing additional military or intelligence support. The loop ends when the battalion commander receives a report that the mission was accomplished.

**3.2 Company (4 platoons, 160 personnel)**

The situation of a company commander is similar to the one of a battalion commander. He has approximately 160 personnel under his command and little possibility to create detailed orders for all of his platoons commanding officers. However from his position it is possible to create a more detailed image of the fragment of the battlefield where the combat will take place. To do that he analyzes several factors like: mission, enemy, territory, troops, time, civilians, space and logistics. All together they are called the METT-TCSL analysis [11]. This analysis is the most time consuming part of this loop so it is done only once during the preparation of the mission concept.

During the combat the commander remains near the main part of his troops so the Su-



**Fig. 6.** The Company Control Loop

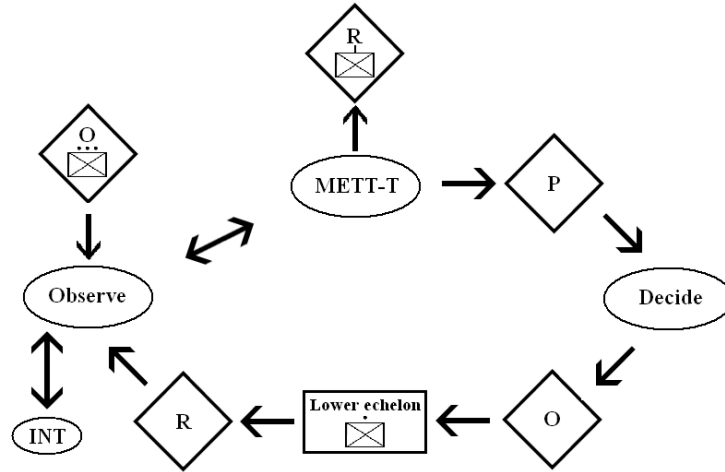
pervision element from the BCL can be substituted with the standard Observe and the

Orient elements.

The Company Control Loop (CCL) will look like the one on the figure 6. The loop starts with receiving the mission. The first Observe element is focused on various aspects of a battlefield and has more a purpose of refining the acquired information taking into account the mission profile. Gathering intelligence is necessary because, although part of the data required for the planning is provided with the order, it is not enough for the commanding officer to create his own, more detailed mission concept. The METT-TCSL analysis takes place and the Mission Concept and the Order are created. The distribution of the orders activates the sub loop. From the Reports and his own Observations the commander creates an image of the battlefield and if necessary issues additional orders. This is being done especially “when the unit’s action must be synchronized with other actions” [12].

**3.3 Platoon (4 squads, 40 personnel)**

On this level of the command the orders have to be detailed enough so that the squads can fulfil their tasks precisely. A platoon leader usually preforms his own METT-TCSL analysis but on a much smaller scale and he focuses on issues important for his platoon. Because of considerably smaller number of personnel that is under his command (in comparison the the company for example) it is much easier to adapt the plan to a changing situation. Therefore only one main control loop with one feedback loop is used here. In order to improve the planning process the feedback is used between the Observe and the Orient stages. The control loop for platoon (Platoon Control Loop - PCL) can be seen on the figure 7. As in the previous loops this one also starts with



**Fig. 7.** The Platoon Control Loop

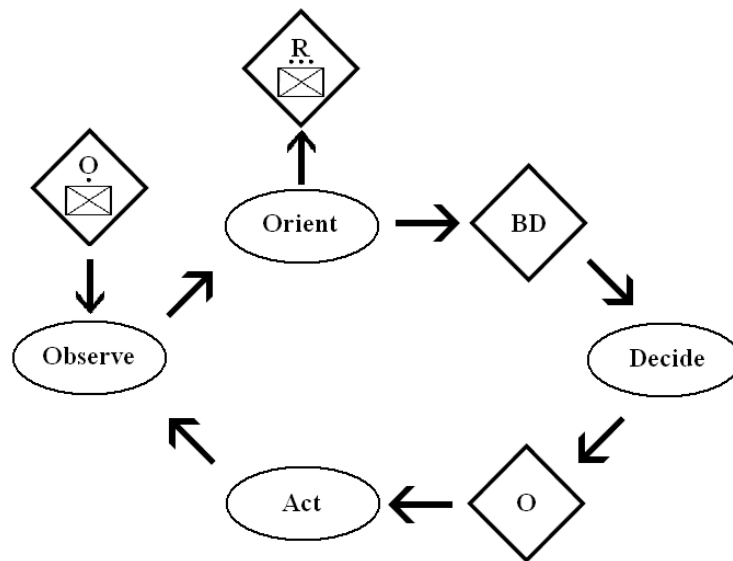
receiving an order from the superior commander. Additional intelligence is gathered

and METT-T analysis is performed. Possible solutions for the problem are developed and the best one is chosen. The orders are transferred to squads which have time to prepare and practice before their execution. During the following iterations the analysis is performed on a much smaller scale, and basically only the deviations from the initial assumptions are considered and proper adjustments to the plan are made. The loop ends when the platoon leader decides that the mission is accomplished.

**3.4 Squad (2 fire-teams, 10 personnel)**

The control loop for squad will be a modified version of the PCL. This is possible because similar methods of command are used on both echelons. The main differences are:

- All the necessary intelligence is gathered by a platoon commander, so squads do not have to do that by themselves
- The plan in this echelon usually consists of standard battle drills (e.g. movement in some predefined formation, known and trained by all soldiers)
- The detailed order that squad receives allows the planning part to be minimized and gives the squad time for preparations
- The squad leader is always in close proximity to the unit so his soldiers' reports can be minimized.



**Fig. 8.** The Squad Control Loop

After taking into account the issues mentioned above the Squad Control Loop can look as on the figure 8. The element that is worth mentioning in this loop is the Act state. Until now the only thing that was passed down the echelon structure was either mission concept or a plan, only on this level the action takes place. Of course the units in a bigger group work together to achieve one general goal, but it is the action at the bottom of the chain of command that makes it possible for the whole machine to work.

## 4 Conclusions

In this paper I presented a new approach towards the decision models that are being used in the military to create command and control systems. After the analysis of the existing decision loops and US Army Field Manuals that describe the C2 process I came to a conclusion that the current approach towards creating decision models should be modified. Although the modern command and control loops focus on the most important aspects of C2 (i.e. observation, orientation, decision and action) they seem to be developed without the overall picture of what are the relations between the echelons in the military forces. As mentioned before, they miss that different levels of command focus on different aspects of C2 therefore should use different types of decision loops. Only after we connect all the loops in one multilayered structure we will have a command and control system capable of describing relations between the levels and providing decisions that will be oriented for a specific echelon. There are several other reasons why I claim the multilayered model to be better than the traditional approach. The first problem that was recognized in all models was the time delay. Making decisions takes time and the C2 systems should give the military the advantage of creating solutions to the problems quicker than the enemy. The multilayered model's advantage in this case is twofold. First—by modeling the whole structure—one will be able to track the delays at all levels of the command and try to find solutions to the problems that can be found at specific echelons. The complete model could also be used to perform an optimization of C2 required at each echelon. The main purpose of this adjustment should be not to increase the level of C2 that we are after, but to decrease the level of C2 that we need [2]. This would cause the desired shift from command by control to command by influence. In practice this means that the level of details in the commander's order and therefore the delay connected to the order preparation will decrease. Second – due to the fact that this structure can give measurable results how fast the decisions can be made and can help to identify existing problems – it might be a very good tool to test existing planning drills. One should remember that the time gained at the top level of the command by using a sophisticated C2 might be wasted by the lower echelons that are using less coordinated methods of planning.

There are also other reasons why the present approach seems to be faulty. It is correct to assume that Boyd's first OODA loop is too abstract and has little connection to the environment. On the other hand we should remember that this loop was developed as an explanation for the superiority of the American pilots over the Koreans. The environment in which it worked was extremely dynamic and there were too many factors that would have to be taken into account in order to create more descriptive version of it. This is similar to the environment in which a modern soldier has to work [4]. In

order to survive his decision making process should take seconds or less. Therefore we know that at the lowest echelon level the best solution is usually the fastest one. So the “bottom” loop should be the fastest and the most general one. On the other hand at the top of the command ladder the art of war and tactics is well described and therefore it is easier to create a set of characteristics that can be analyzed by the C2 system. This argument alone proves that it is impossible to use one general loop to model all command and control activities.

The Multilayered Decision Model (MDM) that I present here requires more research and improvement. It is necessary to create a set of characteristics for each echelon that be used during the initial planning phase. What should also be researched is how to introduce random factors like interferences or deliberate disrupt of command and control process.

This model offers the possibility for further development that should be exploited.

## References

1. Brehmer, B.: The dynamic OODA loop: a new basis for designing C2 support. In: Second International Conference on Military Technology, Stockholm 2005. (2005) 47–56
2. Department of the Navy: MCDP 6: Command and Control. Technical Report PCN 142 000001 00, Department of the Navy (1996)
3. Boyd, J.: A Discourse on Winning and Losing. s. n (1987)
4. Bazin, C.A.A.: Boyd’s OODA loop and the infantry company commander. *Infantry January - February* (2005) 17–19
5. Shewhart, W.: Statistical Method from the Viewpoint of Quality Control. Courier Dover Publications (1986)
6. Finlay, P.: Introducing Decision Support Systems. NCC (1989)
7. Richard Breton, R.R.: The C-OODA: A Cognitive version of the OODA loop to represent C2 activities. In: 10th International Command and Control Research and Technology Symposium, McLean 2005. (2005)
8. David Fadok, James Boyd, J.W.: Air power quest for strategic paralysis. Technical Report AD–A291621, Maxwell Air Force Base AL (1995)
9. van Creveld, M.: Command in war. Harvard University Press (1985)
10. Department of the Army: Field Manual 7-10: The Infantry Rifle Company. Technical Report 7-10, Headquarters, Department of the US Army (2000)
11. Department of the Army: Field Manual 7-20: The Infantry Battalion. Technical Report 7-20, Headquarters, Department of the US Army (2000)
12. Department of the Army: Field Manual 7-8: The Infantry Rifle Platoon and Squad. Technical Report 7-8, Headquarters, Department of the US Army (1992)



# Earned Value Management as Project Follow up

Stefan Lindkvist

Department of Computing Science  
Umeå University, Sweden  
dit03slt@cs.umu.se

**Abstract.** Earned Value Management (EVM) is believed to be an effective project management method in projects. This article shows what the most common problems are in projects. It also show how the EVM can be used to keep the project within schedule and within budget. The article first handle the problems in projects then some history of EVM is given. After this the theory of EVM is explained with different levels of implementations. At the end I going to discuss some problems with EVM. There are also some things mentioned about what is special in software projects.

## 1 Introduction

Earned Value Management (EVM) is a management tool that integrates the technical, cost, and schedule parameters of a contract. During the planning phase a baseline is developed by time phasing budget resources for the defined work. When work is formed and measured against the baseline, the corresponding budget value is “earned”. According to Ernst [1] this is the foundation of EVM. The definition of earned value is according to Software Productivity Center [2]: “*Earned value (EV) is a tracking metric which measures the actual amount of work accomplished, regardless of the effort expended or the time elapsed*”.

Many things can go wrong so that a project falls behind in money or schedule. One of these problems can be as simple as the program that the project is using is not good enough [3]. Likely less than 1% of projects are using EVM, in spite of recognized benefit to this problem [4]. This article will show that if you have the skill to make a spread sheet that is enough to implement an EVM for small projects.

### 1.1 Common problems in software projects

Keil and Robey [5] say that in the Inc Standish Groups survey, written about in "CHAOS: A recipe for Success" from 1999, only 26% of software projects were delivered on time, on budget, and with promised functionality; 46% were delivered behind schedule, with fewer functions and features than originally specified and spent to much money. Escalation of a project occurs when decision makers throw good money after projects that do not have the possibility to succeed [5].

Even if there are evidence of a failing project in the lower ranks of an organization, accurate information about failure may not move up the organization hierarchy, which is called the "mum-effect". The "deaf effect" is when the people responsible for controlling the project refuse to pay attention [5, 6]. Deep insight and understanding of an organizations software process is required in order to identify problems in projects [7].

The first thing that is done in a project is to do a project plan, which means to predict the future, according to Marjan Krasna [8]. The best thing would have been if the plan would be exactly the same thing as the post project measured data. This would be the perfect project plan, but in the end there are always problems. If different parts of an organization use different ways of project planning and different tracking discipline this causes substantial problems [9]. This results is a plan that is late, lacking detail and ultimately unrealistic. Stevenson[10, p.305] says: "*There are only two phases to a big military program: Too early to tell and too late to stop. Program advocates like keeping bad news covered up until they have spent so much money that they can advance the sunk-cost argument; that it's too late to cancel the program because we've spent too much already*". This shows the need for a method to get early warnings when a project is about to go bad.

## 2 Earned Value Management

### 2.1 Background

Earned Value Management (EVM) originally comes from the concept PERT/Cost [11]. PERT/Cost is a way of showing the budgeted project cost based on the activity start times. The assumption behind PERT/Cost is that cost per unit of time for an activity is constant between its start time and its finish time [12]. The problem with PERT/COST was that it was considered inflexible. In the 1960s the idea of EVM took root in the United States Department of Defense (DoD) and 1967 the DoD established a criterion-based approach. The criterion-based approach was using a set of 35 criteria and was called Cost/Schedule Control Systems Criteria (C/SCSC) [11]. In the 1970s and the early 1980s the C/SCSC was often ignored by project managers in both government and industry. C/SCSC was often considered a financial control tool that could be delegated to analytical specialists. In the late 1980s and early 1990s the methodology of EVM emerged, from this point on not only EVM specialists understood and used EVM, but also managers and executives used it. From 1989 when EVM leadership had elevated to the Undersecretary of Defense for Acquisition, EVM became an essential element of program management and procurement. For example, Dick Cheney cancelled the Navy A-12 Avenger II Program due to performance problems detected by EVM in 1991 [13]. In the 1990s, many U.S. Government regulations were eliminated or streamlined. EVM not only survived the acquisition reform movement, but became strongly associated with the acquisition reform movement itself [11]. From 1995 to 1998, ownership of EVM criteria (reduced to 32) was transferred to industry by adoption of ANSI EIA 748-A standard [1, 14]. EVM



quickly spread to the National Aeronautics and Space Administration(NASA), the United States Department of Energy and other technology-related agencies [4].The PMBOK®[15] says that an overview of EVM was included in first PMBOK Guide®First Edition in 1987. Efforts to simplify and generalize EVM gained momentum in the early 2000s. The United States Office of Management and Budget began to mandate the use of EVM across all government agencies, and for the first time, also for certain internally-managed projects (not just for contractors) [11].

The reason to use earned value is to measure how much of the project’s scope and objective that have been accomplished, predict the outcome when completed, using units of measure which are at the core of the value system of the project [1]. Traditionally, the core value has been money, and the focus has been cost. However, less than 1% is using EVM, in spite of recognized benefit [4]. The redesign of the business to overcome the problems with projects falling behind in budget and behind schedual is often impractical and cost has a much lower priority than time or performance in many commercial projects. An example of higher priority element is time-to-market for a new product or service.

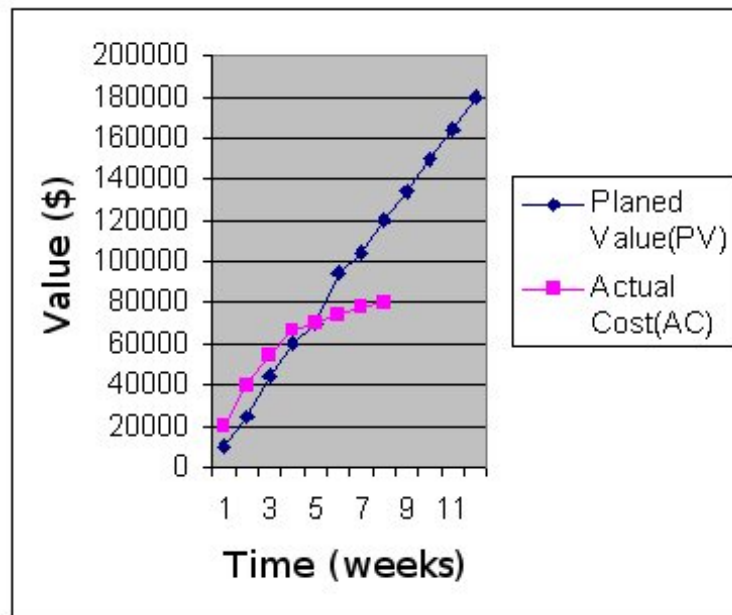


Fig. 1. Project tracking without earned value (redrawn from Fleming [4])

## 2.2 EVM in theory

In Figure 1 is an example of project tracking that does not include earned value performance management. This project has been planned in detail, including a time-phased spending plan for all elements of work. Figure 1 shows the cumulative budget for this project as a function of time (labelled PV) and shows the actual cost of the project through 8 weeks. For those unfamiliar with EVM, it can appear that this project was over budget until week 4 and under budget from week 6 to week 8. Missing from this chart is the knowledge of how much work that has been accomplished during the project. If the project was completed after 8 weeks, then the project would be well under budget and well ahead of schedule. But, if the project is only 10% done after 8 weeks, the project is significantly over budget and behind schedule. What EVM accomplishes is measuring technical performance objectively and quantitatively [16, 1].

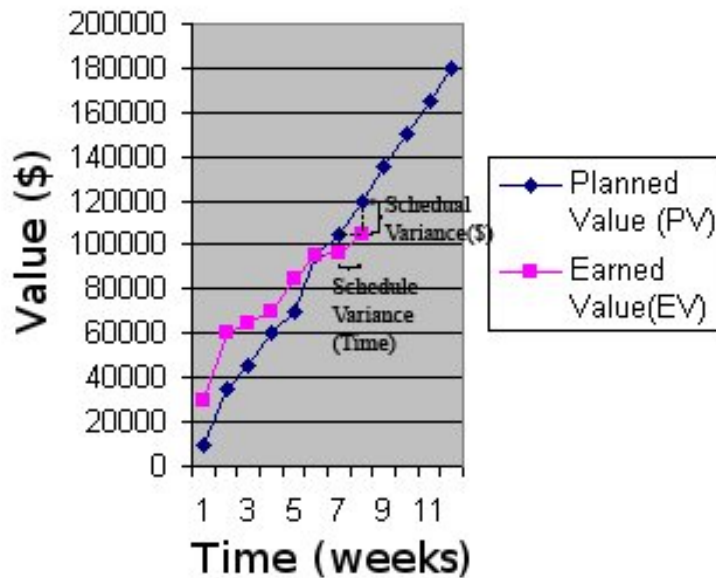


Fig. 2. Earned value with PV

The project plan also includes pre-defined methods of quantifying the accomplishment of work. The project manager identifies every detailed element of work that has been completed each week [1]. Then the project manager summarizes the Planned Value (PV) for each of these completed elements, this is called “earned value” (EV), and it can be computed monthly, weekly, or as progress is made. The next figure, figure 2, shows the EV curve along with the PV curve from Figure 1. This chart shows the schedule performance aspect of EVM. We

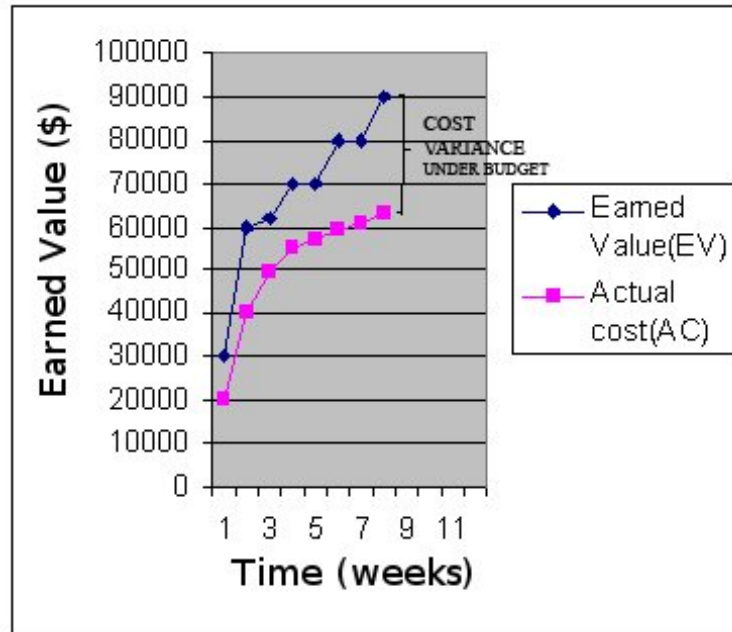


Fig. 3. Earned value with actual cost

can see that the progress started more rapidly than planned, but slowed down significantly and fell behind schedule in week 7 and 8. The EV curve can be plotted with the actual cost data from Figure 1, see figure 3. From this figure it can be seen that since the start of the project it has been under budget, relative to the amount of work accomplished. This is a much better conclusion than the one derived from Figure 1. Our last figure, figure 4, shows all three curves together, this is a typical EVM line chart. The way to read these three-line charts is to first identify the EV curve, then compare it to PV and AC. The foundational principle of EVM can be seen in this illustration, a true understanding of cost performance and schedule performance relies first on measuring technical performance objectively [1].

### 2.3 Different Implementation Types of EVM

Many organizations have established an all-or-nothing threshold; if a project is above this threshold, it requires a full-featured EVM system, but projects below the threshold do not one. For are a small and simple project a “simple implementation” is enough. A big complex project (for the DoD more than \$50M [1]) on the other hand requires an intermediate or advanced implementation [17, 18]. The three different implementations are based on one another and they are described below.

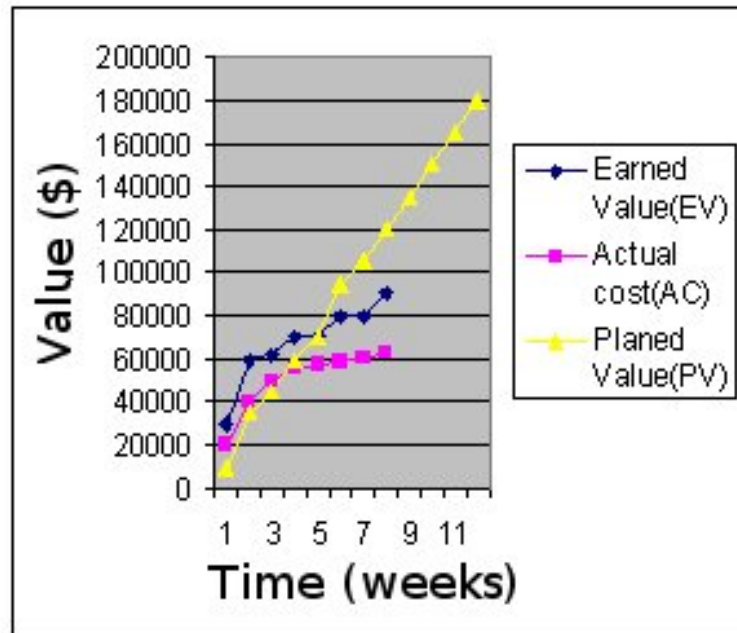


Fig. 4. All graphs in one figure, typical EVM figure (redrawn from Fleming [4])

#### Simple Implementations (emphasizing only technical performance)

Historically only the largest and most complex projects have enjoyed the benefits of EVM, but there are many more small and simple projects. Any person who has basic spreadsheet skills could do a simple implementation of an EVM according to Young [18]. The first step is to define the work. This is done in a hierarchical arrangement called a work breakdown, and in simple projects this can be a list of tasks [17]. The list, or Work Breakdown Structure (WBS), has to be comprehensive and each element in the list has to be self-contained, so that work is easily categorized in one and only one element of work [1]. Figure 5 shows what a WBS can look like. A WBS is used in all of the differed implementations of EVM.

The next thing to do is to assign a value to every terminal element, called the predicted value (PV) [17]. In a large implementation of an EVM (for the DoD more than \$50M [1]), the PV is almost always an allocation of the total project budget, measured in money (e.g., Dollar or Euro), in labour hours, or both. In simple projects each element is assigned a “point value” which does not have to be a budget number. By assigning weighted values a consensus on all PVs can be reached, which is an important benefit of EVM. The process exposes misunderstandings and miscommunications about the scope of the project. The difference between figure 2 and figure 1 is that figure 1 does not say anything about how much work that have been done. The next thing to do is to give each terminal element “Earning rules” [17, 1]. The simplest way to do is to apply

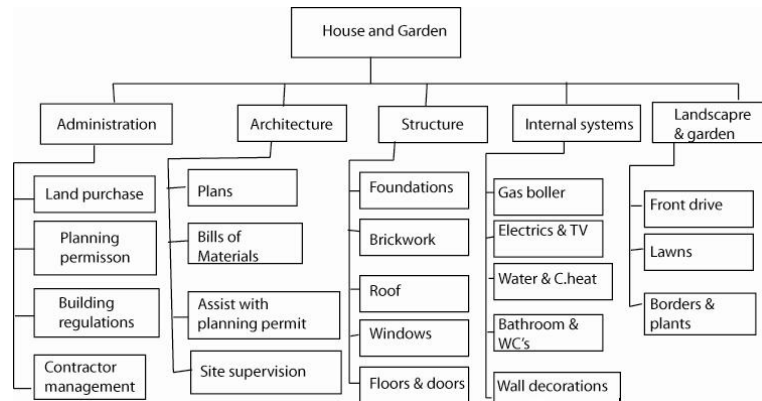


Fig. 5. Work breakdown structure (redrawn from Webb [17])

only one earning rule, like the 0/100, rule to all terminal elements. This means that no credit is earned for an element of work until it is complete. Another related rule is called the 50/50 rule, which means that 50% credit is earned when the work starts on an element, and the other 50% upon completion. These simple earning rules work well for small or simple projects because generally each terminal element tends to be fairly short. The EV is reported regular intervals (e.g., weekly or monthly) or at when work elements are started/completed [1].

One useful outcome of this simple implementation is the possibility to compare EV curves of similar projects, as illustrated in Figure 6. For example, the progress of three construction projects are compared by aligning the starting dates. If these three projects were measured with the same PV valuations, the relative performance of the projects can overtime easily be compared [18]. In Figure 6 the three constructions can be observed, construction No. 2 is a little behind, while No. 3 is a little ahead of construction No. 1.

**Intermediate Implementations(integrating technical and schedule performance)** The intermediate implementation measures schedule performance, following schedule. Schedule performance is equal in importance to technical performance, in many projects. For some new product development projects it is more important to finish quickly rather than what the cost of the project. Finishing after a competitor may cost a great deal more in lost market share [1]. The simple implementation of EVM do not a have timescale for measuring schedule performance so it is not likely that this version of EVM is used in these kind of projects. In intermediate implementations the project manager can implement a critical path to construct a project schedule model. As in the simple implementation the project manager must define the work comprehensively. The project manager starts by constructing a project schedule that describes the links between elements of work. This schedule model can then be used to develop the PV curve, as seen in Figure 2. In addition to the 0/100 and 50/50

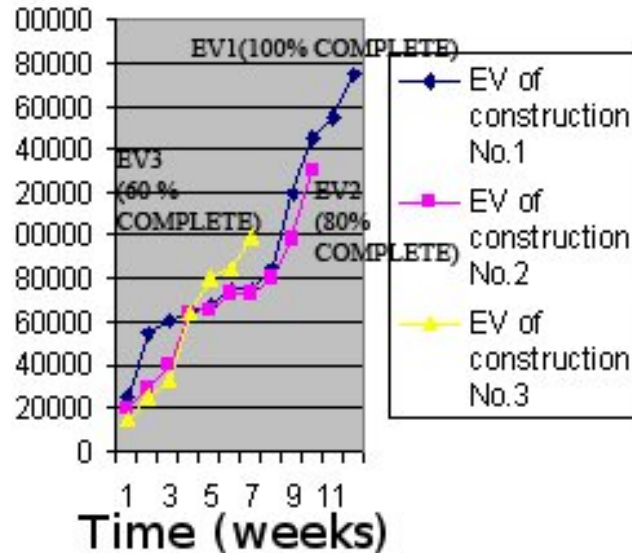


Fig. 6. Compared EV values (redraw from Young [18])

earning rules mentioned above intermediate implementations includes weighted milestones, percent complete estimates, equivalent units, earned standards, apportioned effort, and level of effort (LOE) [18]. Budgeted dollars as one dimension for the PV and EV is common practice, but it is not a requirement [17]. The EVM formulas below are for schedule management, and do not require accumulation of actual cost (AC). This is important because it is not uncommon in small and intermediate size projects for true costs to be unknown or unavailable [17].

**Schedule Variance (SV(\$)):**  $SV(\$) = EV - PV$ , greater than 0 is good (ahead of schedule) [17]

**Schedule Performance Index (SPI(\$)):**  $SPI(\$) = EV / PV$ , greater than 1 is good (ahead of schedule) [17]

**Advanced Implementations (Full-featured) (integrating cost, technical and schedule performance)** In large and complex projects keeping track of cost performance at regular intervals is a requirement in addition to managing technical and schedule performance [1]. To be able to measure cost performance the planned value and earned value must be in units of currency, the same units that actual costs are measured [1]. The planned value curve is commonly called

a Performance Measurement Baseline (PMB). The primary method to delegate responsibility and authority to various parts of the performing organization is to establish control accounts [18]. Control accounts are cells of a responsibility assignment matrix, which is an intersection of the project WBS and the organizational breakdown structure(OBS). Control accounts are assigned to Control Account Managers (CAMs). According to Ernst [1] the United States, the primary standard for full-featured EVM systems is the ANSI EIA-748A standard, published in May 1998 and according to the ANSI EIA-748A Standard [14] refined in August 2002. The ANSI EIA-748A standard defines 32 criteria for full-featured EVM system compliance.

**Budget at Completion (BAC):** The total planned value (PV or EV) at the end of the project. If a project has a Management Reserve (MR), it is typically in addition to the BAC [17].

**Cost Variance (CV)**  $CV = EV - AC$ , greater than 0 is good (under budget) [17]

**Cost Performance Index(CPI):**  $CPI = EV/AC$ , [17]

- $> 1$  means that the cost of completing the work is higher than planned.
- $= 1$  means that the cost of completing the work is right on plan.
- $< 1$  means that the cost of completing the work is less than planned.

**Estimate At Completion (EAC):** EAC is the manager's projection of total cost of the project at completion. ETC is the estimate to complete the project.  
 $EAC = AC + ETC$  [17]

**To-Complete Performance Index (TCPI):** The TCPI calculates what the CPI will be for the rest of the project based on the manager's projection of subsequent performance. The TCPI should be compared to the CPI. Any significant difference should be accounted for to explain why the manager projects either improved or degraded performance in the future.

$$TCPI = (BAC - EV) / ETC \text{ [17]}$$

**Independent Estimate At Completion (IEAC):** The IEAC is a metric for the project's total cost using the performance to date of project overall performance. This can be compared to the EAC, which is the manager's projection.

$$IEAC = \Sigma AC + ((BAC - \Sigma EV)/CPI) \text{ [17]}$$

### 3 Discussion

In this article I have discussed EVM, and how it works. The article has showed how EVM is implemented, but there are much more to learn about it. The article has also shown common problems in projects. There are however problems with EVM too, if the project plan changes rapidly it is difficult to administer [19]. The earned value says nothing about actual value being earned for the organization, therefore EV should model real value as good as possible [19]. It is statistically significant that there is a correlation between if a project is successful and the use of EVM [20]. Today there are many different software projects and they do not look like it in the start of the 1990s, the projects have to be more flexible today [21]. To handle this there have been modifications of the classic EVM, Weaver [21] has one method, Sulaiman [16] and Young [18] have others. With EVM the people that makes the decisions always know how the project is doing and there is no "mum-effect" and partly the "deaf-effect" that Keil and Robey [5] talks about has disappeared. There is probably no perfect project management methodology but I have shown that EVM is good method. In software projects it is hard to measure EV in budgeted money, and even harder when a program is being written. According to Solomon a good way to measure how much of a software-project that has been done is to count functions [22].

### References

1. Ernst, K.D.: Department of defense – earned value management implementation guide (Oct 2006) [http://guidebook.dcmam.mil/79/EVMIGMar2006\\_dcmaevmctr\\_090606\\_B1.doc](http://guidebook.dcmam.mil/79/EVMIGMar2006_dcmaevmctr_090606_B1.doc), accessed 2007-03-15 copy <http://www.cs.umu.se/~dit03slt/Ernst.doc>, accessed 2007-05-16.
2. SPC Resources: Metrics program glossary. <http://www.spc.ca/resources/metrics/glossary.htm>, accessed 2007-05-15 (2007)
3. Economic Development and European Services: Good practice in structural fund project management (2005)
4. Fleming, Q.J.K.: Earned value project management, third edition, project management institute. CROSSTALK The Journal of Defense Software Engineering (Jul 1998) 19–23
5. Keil, M., Robey, D.: Blowing the whistle on troubled software projects. Communications of the ACM **44**(4) (April 2001) 87–93
6. Keil, M., Robey, D.: Turning around troubled software projects: An exploratory study of the de-escalation of commitment to failing courses of action. Communications of the ACM **15**(4) (1999) 63–88
7. Jakob Iversen, P.A.N., Nörbjerg, J.: Situated assessment of problems in software development. The DATA BASE for Advances in Information Systems **30**(2) (1999) 66–81
8. Marjan Krasna, I.R., Stiglic, B.: How to improve the quality of software engineering project management. Software Engineering Notes **23**(3) (1998) 120–125
9. James D. Herbsleb, D.J.P., Bass, M.: Global software development at siemens: Experience from nine projects. In: Software Engineering, 2005. ICSE 2005. Proceedings. 27th International Conference on. (15-21 May 2005) 524–533



10. Stevenson, J.P.: The Pentagon Paradox: The Development of the F-18 Hornet. Naval Institute Press (1993)
11. Abba, W.: How earned value got to prime time: A short look back and a glance ahead (2000) Project Management Institute Seminars and Symposium in Houston, Texas.
12. T. Di Battista, R. Di Nisio, T.S.: Price managing of manufacturing firms using fuzzy cost/pert. [http://www.mtisd06.unior.it/collegamenti/MTISD%202006/Abstracts/80\\_%20Di%20Battista.pdf](http://www.mtisd06.unior.it/collegamenti/MTISD%202006/Abstracts/80_%20Di%20Battista.pdf), accessed 2007-05-01 (2006)
13. Tolley, J.M.: Seminar on metrics "You can't manage what you don't measure" (23-24 Oct 2001) HRA seminar on METRICS.
14. Alliance, E.I.: Ansi eia-748a standard (Jan 2005)
15. Project Management Institute: A Guide to the Project Management Body of Knowledge (PMBOK® Guide). Project Management Institute (Dec 2000)
16. Sulaiman, T.: Agileevm – earned value management the agile way. Agile Journal **07**(2) (Jan 2007) <http://www.agilejournal.com>.
17. Webb, A.: Using Earned Value : A Project Manager's Guide. Gower Publishing Limited (Oct 2003)
18. Young, S.D.: EVA and Value-Based Management : A Practical Guide to Implementation. McGraw-Hill Professional Book Group (Nov 2000)
19. Boehm, B., Huang, L.: Value-based software engineering: Reinventing "earned value" monitoring and control. Software Engineering Notes **28**(2) (2003) 1–7
20. Marshall, R.A.: The contribution of earned value management to project success on contracted efforts: A quantitative statistics approach within the population of experienced practitioners. [http://pmiseminars.org/prod/groups/public/documents/info/pp\\_surveyresults.asp](http://pmiseminars.org/prod/groups/public/documents/info/pp_surveyresults.asp), accessed 2007-03-15 (Oct 2006)
21. Weaver, P.: Earned value business management (20 - 22 Feb 2002) The 6th Australian International Performance Management Symposium.
22. Solomon, P.J.: Using earned value to manage successful software projects (2001)



# An investment perspective on usability

Anders Moberg

Department of Computing Science  
Umeå University, Sweden  
dit02amg@cs.umu.se

**Abstract.** Usability is by its definition something positive for the user. But is it something positive for the investors and the producers? The answer is yes and supporting examples are presented in this paper. The published examples are all presenting positive return on investment from usability design with a few exceptions. Two of these exceptions are presented and with them in mind a shallow but necessary discussion about the findings criticizes the tactical concept of return on investment and suggests that also other measures are being used for example the strategic measure total cost of ownership.

## 1 Introduction

With the background of my university studies focused on interaction technology, interaction design and the user centered design processes I, and many with me, have been taught to think upon usability as something fundamental. No course or lecture focuses on arguments for a usability approach from any other point of view than the users. When I approached the industry and started with my master thesis it was obvious that usability focus was not especially well spread. It was also obvious that the dimension of economy occurred to be missing from the usability courses and at the same time this dimension is fundamental in the industry. There is a great need to present the positive sides of usability not only from the users point of view, but also from the producing and investing companies point of view. Economics would be a well suited language for this presentation.

Within the following definition of interaction design and usability with regard to some of the more fundamental terms of the economic language this article will present a few examples and quotes. These examples show cases where the effect of usability has been measured or estimated in ways that might have a better impact than just proving that usability is good for the user. The examples presented are all of a positive kind where a positive economic effect of usability are stated, this is not biased and subjective because it actually is a representation of the published findings. On the other hand a discussion with only positive voices in it self raises question marks and therefore a discussion of the presented examples will be made. This discussion would be very interesting if empirical data, proving or disapproving, existed but in its absence a common sense evaluation of some of the examples will be preformed.

This paper in its simple form presents collected examples and numbers possible to use as arguments for usability as something positive not only for the user but for various actors of the production chain, from investors to marketers.

### **1.1 Interaction Design**

To define the context in which this article should be read a definition of interaction design would work as a good common ground. Preece describes interaction design as a design of interactive products to support users in both their everyday lives and in their working lives [1]. Another definition by Winograd defines interaction design as the design of spaces for human communication and interaction [2]. It is within this context, the discussion would be targeting usability.

### **1.2 Usability**

Usability and in this case strictly usability of interactive products, is a term for how easy a product is to learn, how effective and how enjoyable the product is to use. One way to define usability is to divide the concept into usability goals. If usability is a cake then effectiveness, efficiency, safety, utility, learnability, and memorability are major pieces or goals of this cake [1]. These goals may be referred to, alone or in some combination, as usability. The purpose of these goals is to use them as a tool when looking upon a part in a process which should be possible to evaluate. For example the efficiency goal leads us to the question of how efficient the software or system solves a task. Does the software need an appropriate amount of the users' attention? These two and many more questions are appropriate in a discussion and evaluation regarding the efficiency goal. For the other usability goals similar questions could be used to evaluate a process and see how well it reaches out towards that specific usability goal.

### **1.3 The economic vocabulary**

The concepts from economics that are fundamental for this discussion are return on investment (ROI) and total cost of ownership (TCO). Return on investment is the ratio between the gain or loss and the investment. The ROI concept is often thought upon as the gained money divided by the invested money but this assumption is not always complete. There are other types of investments than money and there are also other measurements than money. For example the investment could be resources like time, hardware, altruism, reputation and more. The gain could be time, reputation, negative effects on a competitor's sale and so on. The TCO concept is a measurement of the costs of owning the product. In the software business the price and the license fees are parts of the TCO measure but other costs like maintenance, education and training of the users are most certainly even bigger parts in this concept.

## 2 Approach

The approach is a walkthrough of the usability field basically where there is an attempt to discuss usability from a more or less strict economical point of view. The usage of return on investment as a concept has been widely accepted for evaluating the success of resources spent on usability actions both during and after the development. The compilation of articles and statistics, *Cost-justifying usability*, edited by Bias and Mayhew in 1994 [3] has been important for the research in this area.

It seems like the published articles almost exclusively states positive ROI on usability and a lot of other positive effects from usability. This is in a way a great disadvantage for the debate where a balance of negative and positive voices is desirable. One of the few critical voices heard in the discussion of return on investment on usability makes a few points which in a way make it easier to approach this area with an objective mind. This paper will first present some examples and then discuss some of them.

## 3 ROI examples

The most compelling thing with usability and interaction design is that its main purpose is to make the interaction easier, faster, more enjoyable and more effective for the user. If the user who now enjoys the new easier work which is done faster is an employee, the employer must enjoy the more effective worker. So the usability has a positive impact on both user and employer in this simple example. But when it comes to questions like is it economically reasonable to invest a lot of time and money to make software more usable the intellectual process is not as simple as above.

### 3.1 About usability as an area

Usability as a concept and part of the developing process has been identified as a positive factor. There are examples that show increased revenues connected to higher marketability of a product with distinct usability factors as higher end-user productivity and lower training costs [3]. "Usable products also lead to good product reviews. Publications devote space just to this one factor, and good reviews lead to increased sales" [4]. The ideas of high productivity and lower training cost are all desirable for the user and most of all the employers. When usability also is a way to get free or cheap publicity and increased sales the concept gets more compelling for the producer.

A good example is that 63% of a sample of large software projects overran their budgets. When they were asked why their budget estimations were wrong they rated four reasons as highly responsible. The cost for these four reasons could have been reduced by good usability engineering [5]. If this example also applies to other software projects than the ones in the example it is a good argument for investments in usability design.

### 3.2 Where in the process

When discussing the design and development process and usability a possible starting point would be the rule of thumb, that many usability-aware organizations have a cost-benefit ratio of 1:10 to 1:100 [6]. That actually states a ROI between 10 and 100 and that of course is a great advantage for usability-aware businesses. The cost of correcting an error in the development process instead of the early design stages is over ten times higher and the cost to correct errors in released software are over 100 times higher [6]. These numbers, and this rule of thumb simply states that it is unwise to avoid usability in the beginning and instead correct errors later on in the development process or even later in the products lifecycle.

Where in the process the usability resources should be used is a fair question and with the rule of thumb from above the early design stages seems most appropriate. A few examples that also shows in this direction are the following. "The big win, however, occurs when usability is factored in from the beginning. This can yield efficiency improvements of over 700%" [7]. The next example is more detailed. "Changes cost less when made earlier in the development life cycle. Twenty changes in a project, at 32 hours per change and an hourly rate of \$35, would cost \$22,400. Reducing this to 8 hours per change would reduce the cost to \$5,600. Savings = \$16,800" [8]. The following and last supporting example is about a financial service company which had developed an application, and just prior to the implementation when tested on users showed a fatal flaw in the way data was entered, because the assumptions did not compile with the users reality and therefore the application was never implemented [9]. These examples and quotes show us that usability to support error prevention in the early design stages instead of error correction in the latter stages of the process is very cost effective and cooperation with the users in the early stages of the process can prevent expensive or fatal mistakes.

But just concentrating the usability to the beginning of the process might not be the best practice, instead letting the usability perspective be a part of the whole process could be even better. One company applied usability techniques and cut their development time with about a third to a half [10].

An example where simple user tests in the later stages would have been of great importance and would have had a significant cost saving effect is the printer manufacturer that released a printer driver that many users had problem with when they were installing it. Over 50 000 users called the support for help and this cost the company nearly one half of a million dollars. To correct the problem the manufacturer sent out an apology and a diskette with a patch resolving the problem to a cost of three dollars for every sold printer [3].

### 3.3 Profit on usability engineered products

If usability can affect the sales of a product is of course an important factor when planning the design process. Therefore a few examples showing increased results from usability engineering would present some facts. A case study over

a new release of software, where the first version was made without usability engineering and the new release was made with, showed increase in revenue with more than 80% and many users said that the enhanced usability was a key factor for buying the new system [3]. There is also possible to gain a lot out of usable systems inside the own organization. An example is a company which had a usability engineered internal system. The user training cost for this system was one hour; a similar system without usability work had a training period as long as a full week [3].

Websites are in it self a product, e-commerce companies develop or buy software that allows to sell products or services online. "You can increase sales on your site as much as 225% by offering sufficient product information to your customers at the time they need it. One way to do this is to develop product lists that don't require shoppers to bounce back-and-forth between the list and individual product pages" [11]. With this example software for setting up e-shops which are usability engineered can have a higher price than the competitors without the usability aspect or at least use usability as an argument to increase the sales of the software.

### 3.4 Usability as sales argument

There are examples showing that usability, with origins in human factors, has considerable impact on increased productivity in IT organizations. Some interesting number occurs when a major computer company invested 20700 dollar on usability to improve a login in procedure on a system used by thousands of users every day. The improved productivity saved the company 41700 dollar the first day the system was used. On a similar system used by over 100 000 users the same company invested 68 000 dollars and with in the first year the same company gained a benefit of 6 800 000 dollars [3]. This is an impressive sales argument with a ROI of 100 on usability engineered systems. Intranets are an important part of the daily function of companies therefore developers of software and solutions for this might use the following example as a sales argument. Bay Networks spent 3 000 000 dollars and two years to build a model intranet and then saved 10 000 000 dollars each year [12].

Another interesting example where usability or the lack of usability made a company loose income from one of the former most desirable products, the following quote tells the story. "One airline's IFE (In-flight Entertainment System) was so frustrating for the flight attendants to use that many of them were bidding to fly shorter, local routes to avoid having to learn and use the difficult systems. The time-honored airline route-bidding process is based on seniority. Those same long-distance routes have always been considered the most desirable. For flight attendants to bid for flights from Denver to Dallas just to avoid the IFE indicated a serious morale problem" [13]. A good example where the users not only loose in productivity and stops enjoying the work, they actually change their use patterns in an unfavorable way for the service provider.

### 3.5 Usability and e-commerce

The web is a very interesting area for usability, it is a tool and interface many users get in contact with during both paid and unpaid time. So this area is important for research about the gain of the usability concept. Usability improvements on websites are usually large. It is common the usability efforts result in hundred percent or more in increased traffic and sales [14]. Another example with interesting numbers, even though it is from a maybe biased report from the company that made the usability redesign. "... recommendations resulted in significant improvements in customer experience on the move.com Web site. After move.com completed the redesign of the home search and contact an agent features based on Vividence's recommendations, move.com conducted another Vividence evaluation to determine the impact of the changes. In the follow-up evaluation, users' ability to find a home on the site increased from 62 percent to 98 percent" [15]. This is an increase of success with 36 percent units, not as high as the hundreds of percent mentioned above but still a great number.

### 3.6 Summary

The examples presented above all states some kind of positive effect from usability for the investors or the producers. The concept of return on investment is widely used as a measurement. Some of the examples presents numbers to show the positive effect of applying usability in the early stages of the development process one of these numbers say that an effect of 700% in some cases are expectable. Another example presents a project that was terminated because when it just before sharp use was user tested and the users could not manage it, a completely unnecessary loss of the invested money, time and effort.

When focusing on the usability goals like learnability there is an example of where users only needed an hour to learn a well designed system and a full week for a similar but not usability designed system. Another powerful example is the major computer company which invested 20 700 dollars on usability to improve a login procedure on a system with thousands of users and on the first day they saved 41 700 dollars.

## 4 Discussion

The examples above together generate an all positive image of usability from an investor and producer point of view. This might seem to be non-objective but the published examples are all positive with just a few exceptions. One of these exceptions is a pilot study made by the University of Huston. The test was to let participants with usability related jobs estimate the cost for usability testing and then estimate the percentage of time or money this testing would save or add to the project. The project was a scenario for usability testing documentation. This study failed in its intent to prove ROI that could have been used to convince otherwise unwilling management to initiate usability studies [16].



Another exception is an article called “The myths of usability ROI” [17] where the author attacks the concept of ROI in the context of usability and presents six so called myths. “There is a lot of empirical data supporting the ROI claims for usability” [17]. This is one of the myths and the article presents a few examples to support why this is a myth. For example there are a lot of articles published in this area but many of them are based on the same and pretty old empirical data. One reason for this lack of data may be the legal departments of the companies that might stop all numbers regarding development costs and other numbers that are considered as corporate secrets.

#### 4.1 Critique

A previous example presented that in a measurement 63% of a sample of large software projects overran their budgets and when they were asked why their budget estimations were wrong they rated four reasons as highly responsible. The cost for these four reasons could have been reduced by good usability engineering [5]. This data just focuses on what is going on inside the producers’ cycle and how usability could affect the projects budget and therefore the price to the end costumer. This price on the other hand is just one of the costs for the costumer. Maintenance and education are most certainly much bigger posts. Usability engineering should also affect the learning time, the efficiency, the memorability, safety and other perspectives of the software which this example does not even try to measure even though it will affect the investor and producer in the way of more satisfied costumers, longer relationship and a positive spin. So the myth “ROI calculation from the producer’s perspective is sufficient” [17] helps to create a wider perspective on this example.

There are often other factors than just usability that changes the outcome of a redesign or a good development process. Let us discuss the example which stated a possible increase of sales on a site with as much as 225% by offering sufficient product information to the costumer at the time they needed it instead of clicking back and forth between product pages [11]. This example is somewhat old and the examples they used to compare with must have been even older or at least the same age. So the time might be an important factor here. If the number of users of the internet had increased between to the two points of measure what effect may that have on the numbers? Had the users of the internet become more comfortable with online shopping and less afraid of using their credit cards online during this time? Is it possible that with respect to these questions the isolated positive effect of usability was small or at least smaller than the stated 225%.

It is actually possible that a usability redesign might result in disadvantage for the user and not the expected more than 100 percent increase presented in one of the examples. If the old non-usability designed site follows a bad but consistent standard the user might actually get lost and feel unsafe in a new design. A scenario could be if a design team discovers that replacing the footbrake in a car with a sensor listening for the word brake or stop. This might be the

best affordance, easiest to understand, easiest to remember but it is still non-consistent with other cars so a driver might feel insecure and lost and instead prefer an ordinary car. The same could happen to a redesigned website. So it is possible that the release campaign and the fresh looks of the site is what pushed the sales and traffic and the new smart usability factors actually suppressed the sales a bit, not likely but still very possible.

There is a big problem if the numbers from these examples are used with out care. "Consider that there are one billion users on the Internet. Assume that only half of them come to your site, but, of these, 80 percent leave without buying. If the average cost of an abandoned shopping cart is \$20 you will lose \$8 billion a year in sales. With a \$5,000 heuristic review, you could reduce abandoned shopping carts by at least 50 percent, thus increasing annual revenues by \$4 billion, an ROI of 80,000,000 percent" [17]. This quote is a sarcastic example but it shows upon something that must be discussed. The example called the rule of thumb that states a 10 to 100 times reduce in costs with a usability perspective early in the process [6]. This ratio could carelessly be used like this. If a software project with a budget of 100 000 dollars actually could reduce this to 10 000 or just 1000 just by applying usability design then it actually seems a bit over optimistic. A great carefulness is advised when using these numbers.

One example talked about a project which was developed without thinking about the user and when the first and only user test was conducted the users could not use the software [9]. This is a common thing in almost every project with a costumer. One of the first thing that should be done when thinking about investing time and money in a product is to see if there is anyone prepared to pay for it. In economics this is a called market research or preliminary market research and is fundamental for good business. With this in mind a usability focus was not the only factor that could have been saving this project, a better management in strict economical point of view could as well have had the same effect.

#### 4.2 Another measure

It seems like the concept of ROI is not complete enough to measure the economical benefits of usability. The critique above expresses that ROI just measure from projects start to the delivery of the software but the greatest benefit of usability occurs to the end costumer. If it is possible to significant reduce the expenses for example, training and maintenance that should affect the producer in a positive way. The concept of total cost of ownership is presented as another measure [17]. If the training time is reduces and the maintenance costs cut then the total cost of owning and using that software is lower. This measure supports the decision makers to understand the impact usability will have on the costumers. This impact is easy to translate in pricing, advantages against competitors, important sales arguments and marketing advantages. With other words ROI is tactical and the TCO is strategic [17].

The TCO value could be estimated to some extent with limited resources. The following scenarios are made up without any empirical data but however

they will present a way to use the TCO measure. A system has gone through a redesign and the functionality before and after is all the same. This system is used for one task in particular of thousands of users on a big company. The task is to log on to the intranet. This is done by clicking on a icon on the desktop of the computer and then write the employee number and a password. In the new version this program auto starts and the employee number is remembered since the last time. This redesign reduces the time for the procedure in average from 20 to 12 seconds. With four thousand users logging on twice a day this is a reduced cost with more than 17 hours and that actually represents the total work time of 2 employees a day, a significant reduction of the TCO. An even simpler example is a system that has an in average two days long training period, free support and maintenance during a three year period. This is simple for a costumer to understand and translate in to actual costs for their company and than balance that against the old system and systems from other companies.

## 5 Conclusion

I would like to start this conclusion with a quote from Donald A. Norman. "Designing well is not easy. The manufacturer wants something that can be produced economically. The store wants something that will be attractive to the costumers. The purchaser has several demands. In the store the purchaser focuses on price and appearance, and perhaps on prestige value. At home the same person will pay more attention to functionality and usability" [18]. This quote exemplifies the many demands on designer and the important but not unique need of usability from the users' point of view. The producers' economic point of view is the main perspective through out this article. If not the strict economics, then at least arguments that can affect decision makers who often is deciding based on economic incentives.

When summarizing this paper the following statements must be done. There are documented cases of positive return on investment and also almost none with negative examples. The concept of return on investment is criticized because it only measure within the producing company and is therefore only a tactical measure and misses the often strategic values of usability design as reduced maintenance cost, reduced learning time and satisfied users. Parts of these strategic values can be measured with the concept of total cost of ownership.

By using both tactical and strategic data a more complete and better founded decisions could be made and therefore I would recommend the future use of both ROI and TCO and even other metrics. The more data and examples there are published strengthens the arguments for usability design even from an economical point of view and in the long run this should lead to much better designed software and websites.

## References

1. Preece, J., Preece, J., Rogers, Y., Sharp, H.: *Beyond Interaction Design: Beyond Human-Computer Interaction*. John Wiley & Sons, Inc., New York, NY, USA

- (2001)
2. Winograd, T.: From computing machinery to interaction design. In: *Beyond Calculation: The Next Fifty Years of Computing*. Springer-Verlag (1997)
  3. Bias, R.G., Mayhew, D.J., eds.: *Cost-justifying usability*. Academic Press, Inc., Orlando, FL, USA (1994)
  4. Marcus, A.: Return on investment for usable user-interface design: Examples and statistics. Technical report, Aaron Marcus and Associates (2002)
  5. Nielsen, J.: *Usability Engineering*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
  6. Gilb, T.: *Principles of software engineering management*. Wokingham : Addison-Wesley (1988)
  7. Landauer, T.K.: *Trouble with Computers: Usefulness, Usability, and Productivity*. MIT Press, Cambridge, MA, USA (1996)
  8. Human Factors International: (-) <http://www.humanfactors.com/about/finance.asp>, accessed 2007-05-08.
  9. Dray, S.: The importance of designing usable systems. *interactions* **2**(1) (1995) 17–20
  10. Bosert, J.: *Quality Functional Deployment: A Practitioner's Approach. Cost Justifying Usability* (1994)
  11. User Interface Engineering: (Are the product lists on your site reducing sales?) <http://www.ue.com/publications/whitepapers/PogoSticking.pdf>, accessed 2007-05-08.
  12. Fabris, P.: (You think tomatoyes, i think tomatoyes) <http://www.cio.com.au/index.php/id;1966828119>, accessed 2007-05-08.
  13. Cooper, A.: *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. Sams (2004)
  14. Nielsen, J.: (Web research: Believe the data) <http://www.useit.com/alertbox/990711.html>, accessed 2007-05-08.
  15. Vividence: (Move.com improves customer experience) [http://www.vividence.com/resources/public/solutions/Success\\_Stories/movecom.pdf](http://www.vividence.com/resources/public/solutions/Success_Stories/movecom.pdf), accessed 2007-05-08.
  16. Ostrander, E.: Usability testing of documentation has many benefits of unknown value (2002) <http://www.stcsig.org/usability/newsletter/0010-pilotstudy.html>, accessed 2007-05-20.
  17. Rosenberg, D.: The myths of usability ROI. *interactions* **11**(5) (2004) 22–29
  18. Norman, D.: *The design of everyday things*. Basic Books (2002)