Predicting Context Aware Computing Performance

By Eli Rohn

Forecasting the growth and key success factors of context aware computing is an area that is wide open for further research.

Abstract

Computer systems will increasingly need to be sensitive to their context to serve their users better. In this paper we forecast the capabilities of context aware computing and relating it to a time frame. Our short-range analysis indicates such systems will become commercially available and common by 2007. Our mid-range analysis indicates that the development pace of Context Aware systems will start to slow down by 2020 and will reach maturity by 2035.

1 Introduction

The purpose of this paper is to forecast the capability of context aware computing and relating it to a time frame. Forecasting can be useful for decision-makers, for techno-social watch groups, and even for the uninitiated who invest in the stock market. We use available information to evaluate such systems' performance on a scale developed for this purpose. We use the data to calculate a fit to a Pearl curve, also known as the Logistic curve or the natural growth curve. This paper is organized as follows: in part two we give an overview of context aware computing, followed by a brief refresher on the Pearl curve in part three. In part four we review existing context aware systems and assign them a value on a ratio scale we developed. The fifth part is dedicated to calculating a fitted Pearl curve, and the sixth part analyzes the result. The seventh and last part concludes the article with a few comments about the vulnerabilities of forecasting and thoughts about further research.

2 Context Aware Computing

Before we discuss Context Aware systems, it is worthwhile to take notice of their precursors, namely Decision Support Systems (DSS). Such systems appeared in the early 1970's. In its infancy, DSS supported only individual decision-makers. Later, DSS were applied to workgroups and teams [1]. However, the term DSS and its offshoot, Executive Information Systems (EIS), have all but disappeared from vendors' web sites and a new generation of systems has emerged, namely the Business Intelligence (BI) applications.

> "There is an increased recognition that business intelligence, which integrates many of the EIS, DSS and expert systems concepts, is becoming a necessary component in the second generation of enterprise systems… The expert systems technology is now being replaced by a diversity of intelligent systems, which are built to fulfill two key functions: (i) the screening, sifting and filtering of a growing overflow of data, information and knowledge (described above) and (ii) the support of an effective and productive use of information systems, which quite often are tailored to the needs and personality of the user. The intelligent systems, which can be implemented for these purposes, range from self-organizing maps to smart add-on modules to make the use of standard software more effective and productive for the users." [2].

A system is context-aware if it can extract, interpret and use context information and adapt its functionality to the current context of use. Context Aware Computing is a field that receives a lot of attention in research and development. Some limited context aware systems have been trickling to the defense industry and now we begin to see limited context aware applications entering the general market. Thus, exploring theoretical limits is useful not only for academics, but foremost for practitioners, corporations, government agencies, and society as a whole.

> "A growing realization is that computer systems will increasingly need to be sensitive to their context. Traditionally, hardware and software were conceptualized as input-output systems: systems that took input explicitly given to them by a human, and acted upon that input alone to produce an explicit output. Now, this view is being seen as being too restrictive. Smart computers, intelligent agent software, and digital devices of the future will have to operate on data that is not explicitly given to them, data that they observe or gather for themselves. These operations may be dependent on time, place, weather, user preferences, or the history of interaction. In other words, context."[3]

Defining what constitutes context is a task by itself. In elementary school composition classes we learn to address key context questions, namely the "6-W": Who, Where, When, What, Why, How. One may argue that context is provided by Identity, Location, Time, and Activity. Liberman and Selker [3] suggest that "context is everything but the explicit input and output" to a system.

Context Aware Computing is not a monolithic area of research and development. While we do not treat each type independently in this paper, it is important to understand that Context Aware Computing becomes progressively more complex as we move from passive applications to active applications. Context Aware systems differ from other software in a number of ways. Such system can be:

- *Adaptive systems*: these learn their user's preferences and adjust accordingly [4].
- *Responsive systems*: these anticipate the user's needs in a changing environment.

- *Proactive systems*: these are goal-oriented, capable of taking the initiative, rather than just reacting to the environment.
- *Autonomous systems*: these can act independently, without human intervention.

Context is an illusive concept. For example, there is no one agreed upon definition, there is no understanding how much “related data” is enough to be recognized as sufficient context, there are challenge in the complexity of capturing, representing and processing contextual data. Key features for that task are information granularity, organization and causation. There is a challenge in determining what constitutes a “similar” situation; to what degree are two events similar, what are the key differences, to what degree differences limit applying a solution that’s good for one situation to a similar one? For example, is swimming with dolphins similar to swimming with sharks? Does it matter if the swimmer is a marine life researcher or a vacationer who entered the same water? Life experience indicates that we humans have limitations in recognizing, understanding, and applying context. Intuitively it seems that context is linked to judgment and applying it depends on “common-sense".

Context Aware Computing relies on several independent enabling technologies. For input it relies on sensors, and these have their own limitations. For processing it relies on hardware, it depends heavily on artificial intelligence principles and those depend on specialized software for implementation, which in turn, relies on data and rules extracted from knowledge of the world, which probably includes knowing specific user’s preferences. Hence, a key-enabling feature is the ability of computer systems to learn from past experience, and use that knowledge in future computations. One area that needs to be formally developed is the concept of similarity. A key feature to the realization of Context Aware Computing is the ability for a system to have “common sense”, at least to a degree that the user would consider it as such.

The aforementioned issues suggest that context aware computing has some limits. In this paper we examine one theoretical limit, utilizing forecasting techniques. In addition, we contribute a list of context aware devices that seems to have escaped surveys of work done in this field. We also propose (a somewhat naïve) comparison scale for such systems.

3 Logistic Formula Refresher

There exists a similarity between the behavior of biological growth and the growth in performance of technological devices. Such growth has been described in equations and graphs. The Pearl curve, also known as the *logistic growth curve* is attributed to Raymond Pearl (1870-1940), an American biologist and demographer. The equation for the Pearl curve is:

$$y = \frac{L}{1 + ae^{-bt}}$$ **Equation 1** Pearl Curve

In this equation, L is the upper limit to the growth of the variable represented by *y*, *t* is time and *a*,*b* are parameters. Plotting the equation yields an "S" shape curve. This curve differs from the geometric curve in two ways: (a) It has an upper asymptote (the curve does not exceed a certain maximum level), and (b) it approaches this asymptote smoothly, but abruptly. It has its inflection point in the middle, thus the upper and lower halves are mirror images of each other. Due to the symmetry "*a*" determines where the curve will be located on the time axis, and "*b*" determines the steepness of the rising portion.

The values of "*a*" and "*b*" are determined by using historical data points and then calculating a curve-fit to the data. This is obtained by a least-squares fit calculation. The equation doesn't lend itself to easy regression calculation; hence we transform it as follows:

Let $\frac{L}{y} - 1 = ae^{-bt}$ **Equation 2**

Then we let $Y = \ln\left(\frac{L}{y} - 1\right) = \ln a - bt$ **Equation 3**

From the historical data we obtain a value Y_i corresponding to times t_i now it is possible to carry out the minimization of

$\sum_{i}^{N} (Y_i - \ln a + bt_i)^2$ **Equation 4**

to obtain a parabolic regression fit for Y. This procedure minimizes the sum of squares of the ratios of the original data and the fitted curve. Equation 4 will be used later in this paper to calculate the Pearl curve for Context Aware Computing.

In peer-reviewed publications and in books, the equation has been successfully used in the analysis and prediction of technology. For instance, steam engine efficiency, power plant efficiency, penetration of telephones to households in the US, efficiency of illumination, productivity of aircrafts (in passenger-miles per hour), top speed of US combat aircraft, time required for inversion of a matrix by a digital computer, ratio of RAM size and access time, to name a few. This suggests the Pearl curve may be appropriate in analyzing some context aware computing theoretical limits.

4 Context Aware Computing Progress

To calculate the Pearl curve for Context Aware Computing we need to review progress in the field and assign a performance value to it. Unlike mechanical devices whose operating parameters are relatively easily quantified, quantifying the degree of context awareness in a device is, at this point, a partially subjective evaluation. We suggest a ratio scale ranging from 0 to L, where 1 means very little context awareness, and L means infinite context awareness. This exceeds human capabilities.

The scale has three parameters: the number of inputs to the context aware system, the number of outputs, and subjective "sophistication" bonus points. The three are summed to yield a value for the degree of context awareness a given system exhibits.

The first generation of context sensitive machinery were electrical analog gun directors [5]. It is historically interesting to note that Bell Laboratories digital computers were used during World War II in the design and testing of these gun directors. We assign the electrical analog gun directors application the value of "3" in our context-sensitivity scale. These were followed by electro-mechanical computers attached to anti-aircraft guns in mid 1955. We assign such applications the value of "4" in our context-sensitivity scale. At about the same time, some tests were conducted by NACA using radar-controlled sights for military aircraft. "Both an optical disturbed-reticle sight and a scope presentation director system with automatic radar tracking were evaluated" [6]. We assign such applications the value of "6" in our context-sensitivity scale. It is of historical interest to note that modern fire control systems are capable of fully automated and autonomous tracking, loading, aiming, and guidance. The next improvement came on board of military fighters and bombers (circa 1960). Specifically, these were sights for delivery of diverse ammunition in a variety of flight profiles. Their context was mainly ammunition type, target type, winds, aircraft motion, aircraft attitude, and aircraft load factor ("G" force). These systems used analogue computers, a serious rival to digital computers in some applications. We assign such applications the value of "8" in our context-sensitivity scale. The next generation had radar added on board to track mobile or air-born target maneuvers. We view the addition of such a sensing device as analogous to growth from self-awareness to social awareness. We assign such applications the value of "10" in our context-sensitivity scale. We can plot one more point in the progress of context aware computing in 1972. NASA's Dryden Flight Research Center had the Fly-By-Wire Project on its way. With a Fly-by-Wire system a pilot moves stick and pedals, transmitting commands to an onboard digital computer. The computer decides on magnitude of response and simultaneously controls other systems such as thrust (engine output) and secondary control surfaces such as air brakes and flaps. This allows building aerodynamically unstable aircrafts, capable of unprecedented maneuvers. However, the pilot isn't in direct control of the aircraft. The Fly-by-Wire system is sensitive to the aircraft's capabilities and configuration at the time input is received from the pilot. It makes the ultimate decision on how to respond, if at all, to the input that reflects the pilot's decision.

"By replacing cables, linkages, push rods, pull rods, pulleys, and the like with electronic systems, digital fly-by-wire reduces weight, volume, the number of failure modes, friction, and maintenance. It also enables designers to develop and pilots to fly radical new configurations that would be impossible without the digital technology" [7]. However, there is one documented accident from 1988 where the pilot of a US F-18 military aircraft failed to recover from a spin because the on-board computer thought his commands were too extreme and blocked them [8]. The August 1993 Gripen fighter crash at an air show in Stockholm is also attributed to the onboard fly-by-wire system [9]. We assign NASA's application the value of "11" in our context-sensitivity scale. By 1980 Fly-by-Wire systems have been implemented in all advanced military fighters and bombers. We assign such applications the value of "16" in our context-sensitivity scale. In the mid 1990's such systems trickled to the civil aircraft industry. They too are assigned the value of "16" in our context-sensitivity scale. All modern air carriers are controlled using fly-by-wire systems. Some aircrafts have a hydro-electro-mechanical backup system. Car manufacturers are currently experimenting with drive-by-wire systems for future cars [10]. In 1995 or thereabout a classified Autonomous Aerial Vehicle by the name of Predator was commissioned. Some models of the Predator allegedly have the ability to roam over enemy territory, and autonomously select targets and destroy them. The Predator flew at least 600 missions in Bosnia. The authors were not able to find reliable information about the nature of the missions or their outcomes. Predators flew quite a few missions over Afghanistan in the chase after Al-Quaida. The CIA used a Predator to fire a missile and blow up a car carrying six suspected Al-Quaida operatives in Yemen on November 2002 [11]. We assign the Predator the value of "30" in our context-sensitivity scale.

The aforementioned systems lack the ability to learn from experience, and are not configurable for a specific user. One may argue it is not necessary for this class of systems to have such attributes. Yet, we believe that full-blown context aware systems should tailor themselves to personalized preferences by learning from experience, in a non-intrusive manner.

In 1992 Olivetti developed one of the first context-aware civil applications, the Active Badge system [12]. The location of persons wearing a special badge transmitting location data using infra-red in an office was known. Their incoming phone calls were forwarded to the closest phone in their location. We assign Active Badge the value of "3" in our context-sensitivity scale. In 1993 Xerox developed the ParcTab, one of the very first palmtop devices with software that acted as a mobile personal digital office assistant. Some of it was context aware augmented. It presented information pertaining to the user's location in the building, helped locate nearest printer and other similar resources, show files associated with a specific room or location when user entered the location, and help in locating other users with the ParcTab. We assign Active Badge the value of "6" in our context-sensitivity scale. Forget-Me-Not was an application added in 1994 to ParcTab for the purpose of recording its user's activities using notes and timestamp. We assign Forget-Me-Not the value of "7" in our context-sensitivity scale. In 1995 the CyberGuide project explored a prototype of a mobile context-aware tour guide. Its investigates how portable computers can assist in exploring physical spaces and cyberspaces[13]. We assign CyberGuide the value of "3" in our context-sensitivity scale. In the same year Stick-e Notes was invented as the electronic equivalent of Post-it notes [14]. A Stick-e Note can be attached to a location or an event and invoked conditionally only when certain criteria were met. We assign Stic-e Notes the value of "6" in our context-sensitivity scale.

In 1996 a system named Wearable Remembrance Agent came into being[15]. The system runs continuously, proactively providing hints based on the physical context of the device. It can provide notes that could be relevant in the user's current context. We assign this Agent the value of "7" in our context-sensitivity scale. In 1997 the CyberDesk project team published data on their work [16]. The CyberDesk uses informational context to change the set of resident services and offer relevant suggestions to the user. Using informational context to automate service integration the system integrates computerized user services. Informational context refers to any artifact (e.g. words on a screen or a picture at a museum) that a user is attending to. Examples of user services are: An e-mail browser, web-based map service, contact manager. For instance, a user can be reading an e-mail message that has information (Web address, e-mail address, routing, time sent etc.) on a new book written by a favorite author. The user highlights the e-mail address, and the system gives him some suggestions on what he can do: search on the author's name, save the contact information, call the author, or send an e-mail. We assign CyberDesk the value of "15" in our context-sensitivity scale. In 1998 StartleCam became public [17]. Using mobile computing, a camera, and a skin sensor, the camera takes images automatically when the system senses "excitement" via the skin sensor. The skin's conductivity signal changes, and this is correlated to arousal levels. We assign StartleCam the value of "7" in our context-sensitivity scale.

The year 1999 seems to be very fruitful in the number of context aware related projects. Georgia Institute of technology had at least three [18] context aware implementations. The In-Out board gathers entrance and exit information about participants wearing a uniquely identifiable iButton who enter and leave the premises at its single entry point. The data can be displayed on any computer that has access to the system. We assign the In-Out Board the value of "4" in our context-sensitivity scale. A user enters some personal information to The Conference Assistant [19] which in turn then displays the conference timetable, highlighting events of interest to the user. When entering a room where a presentation is given, the Conference Assistant retrieves and displays information about the presenter. If presentation material is available, the user can access it and annotate it. The annotated material data is recorded with time stamps. We assign The Conference Assistant the value of "5" in our context-sensitivity scale. The system nicknamed DUMMBO (Dynamic Ubiquitous Mobile Meeting Board) is an instrumented digitizing whiteboard, which supports the capture and access of informal and spontaneous meetings. Its inputs are Location, nearby participants Ids, Audio, Image, timestamps. It stores the whiteboard drawing, conversation audio, and timestamps. It is sensitive to people joining or leaving a predefined perimeter near the whiteboard. We assign DUMMBO the value of "8" in our context-sensitivity scale. Smart Sight [20] is a tourist assistant uses a GPS to determine its location along with some data about the user as its context information. It is intended to translate from and to local language, to provide information about landmarks, and to keep track of places the user marks as interesting. We assign Smart Sight the value of "9" in our context-sensitivity scale.

Circa the year 2000 HP revealed its vision of a technology future as demonstrated by the CoolTown project. In this vision, devices and services are federated and are context-aware; everything and everybody has a web presence, thus enabling the connectivity that is required for some types of context aware applications [21]. We assign CoolTown the value of "17" in our context-sensitivity scale. In 2001 Microsoft gave the world a glimpse of a context aware project it is running. Microsoft's EasyLiving is a ubiquitous computing project of the Vision Group at Microsoft Research. It concentrates on smart homes and smart offices [22].We assign EasyLiving the value of "20" in our context-sensitivity scale.

Now that we have organized and quantified the progress of Context Aware Computing, we are ready to move to the next step: Fitting the data to the Pearl curve and analyze the results.

5 Fitting Context Aware Computing to the Pearl Curve

The calculation has several steps: setting up historical performance, calculating y for each historical point, converting it to LOG10(y), calculating the predicted (fitted) value for each y, calculating the intercept ("-A") and the slope ("B") of the curve then plugging those values back into the Pearl formula to draw the fitted curve.

The intercept is the point at which a line will intersect the y-axis by using existing x-values and y-values. The intercept point is based on a best-fit regression line plotted through the known x-values and known y-values. We use the intercept to determine the value of the Performance (dependent variable) when the Year (independent variable) is set to zero.
The equation for the intercept of the regression line is:

$$a = \overline{Y} - b\overline{X}$$

"b" is the slope of the regression line. It is the vertical distance divided by the horizontal distance between any two points on the line, which is the rate of change along the regression line. The equation for the slope of the regression line is:

$$b = \frac{n\sum xy - \left(\sum x\right)*\left(\sum y\right)}{n\sum x^2 - \left(\sum x\right)^2}$$

Before we plot the fitted curve, we must make a hypothetical assumption about the value of "L", since it impacts the fitted curve.

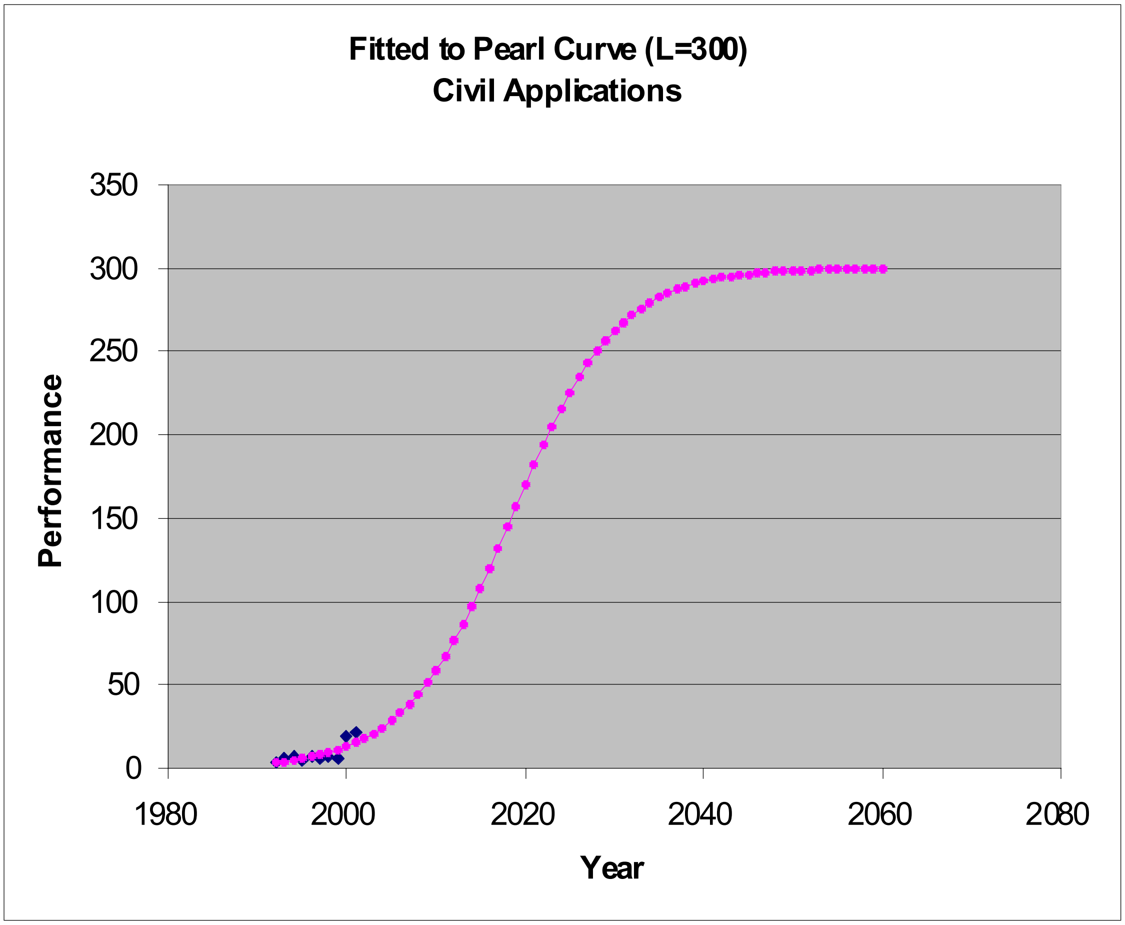
We plot two curves: one assuming L=300 and the other assuming L=3000. For L=300 we assume an expected performance growth of about 13 times by the time growth rate becomes insignificant. For L=3000 we assume an expected performance growth of about 136 times by the time growth rate becomes insignificant. One may argue why these values, and why not expect a growth of one thousand, reaching L=22,000. Let us examine this issue further.

The selection of a hypothetical upper limit is an educated guess. For our purposes, we assume that growth in context aware software would be similar to growth in other areas of complex software. One source such area is Decision Support Systems. In [23] there is a historical overview of DSS evolution since the 1970's. We can use the historical growth and their replacement or evolvement into a new paradigm as a proxy for context aware computing. This is where we get an estimated value of "300". It is also interesting to explore an assumption that allows for ten times that growth. This is where we get the value of 3000. Further, we believe that Knowledge Representation is a key enabler in the advancement of context aware computing. We estimate this field has not exhibited performance growth by a factor of three or more.

For the civil applications we obtained the following data and calculated the predicted (fitted) value for "y":

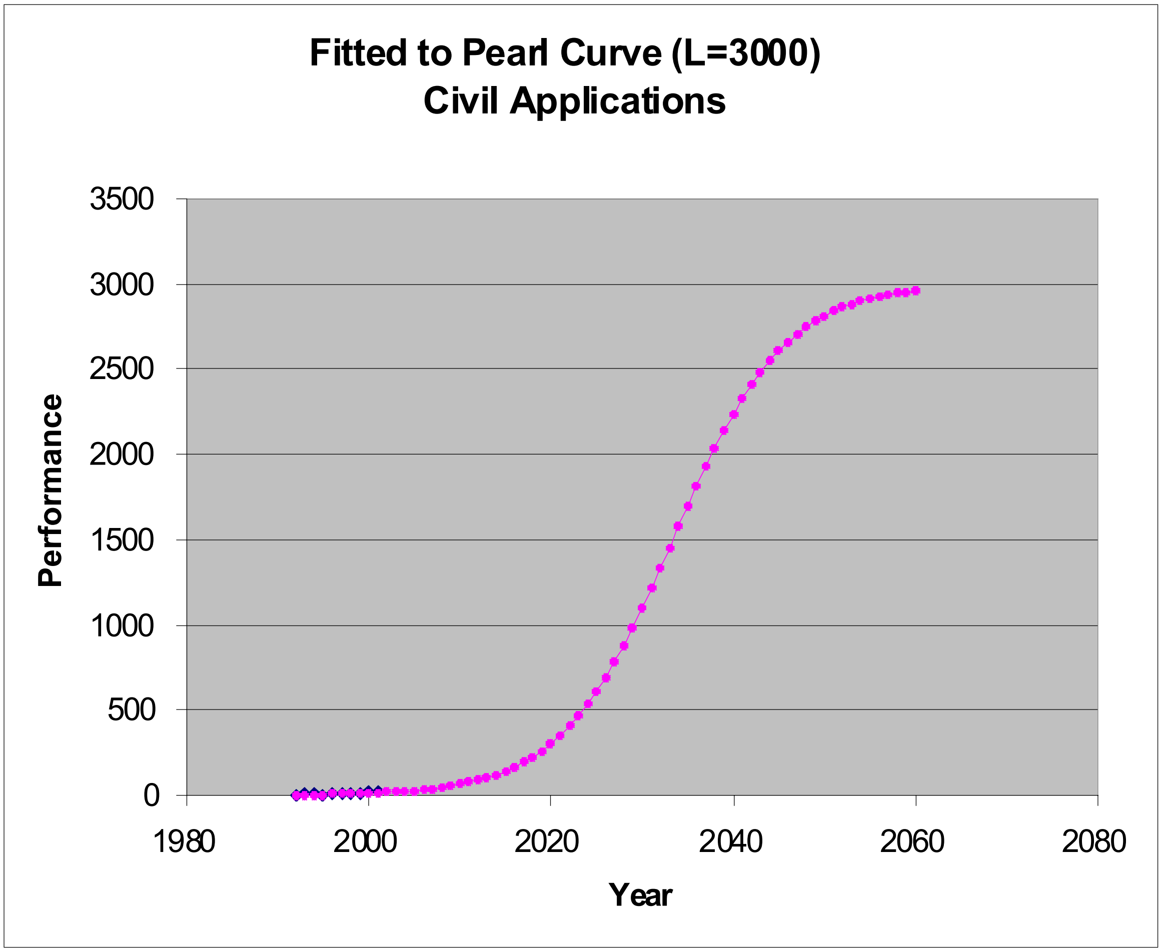
Year	Performance	Y=y/(L-y)	Log10(y)	Predicted y
1992	3	0.0101	-1.9956	3.5084
1993	6	0.0204	-1.6902	4.1416
1994	7	0.0239	-1.6218	4.8870
1995	4.5	0.0152	-1.8173	5.7641
1996	7	0.0239	-1.6218	6.7950
1997	6	0.0204	-1.6902	8.0052
1998	7	0.0239	-1.6218	9.4239
1999	6.5	0.0221	-1.6547	11.0847
2000	19	0.0676	-1.1700	13.0249
2001	22	0.0791	-1.1016	15.2868

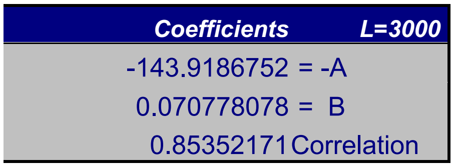
For L=300 we get the following:



Coefficients	*L=300*
-147.3005855	= -A
0.072978758	= B
0.851772224	Correlation

For L=3000 we get the following:

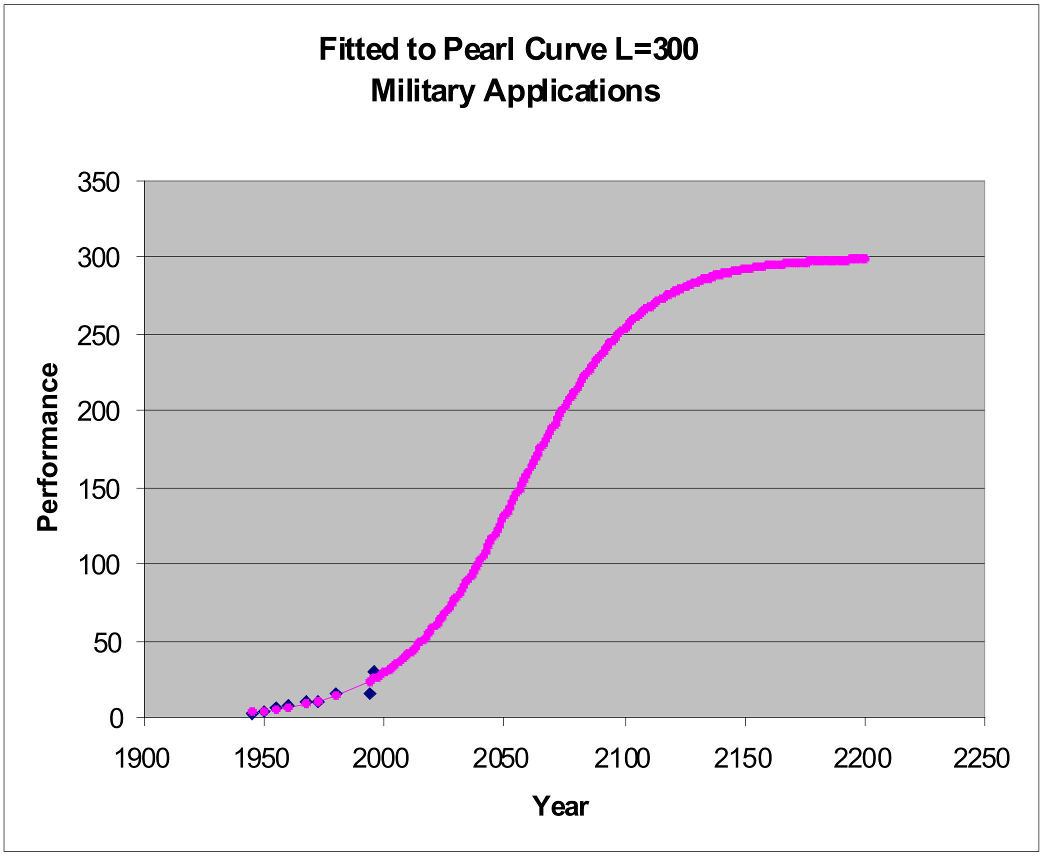




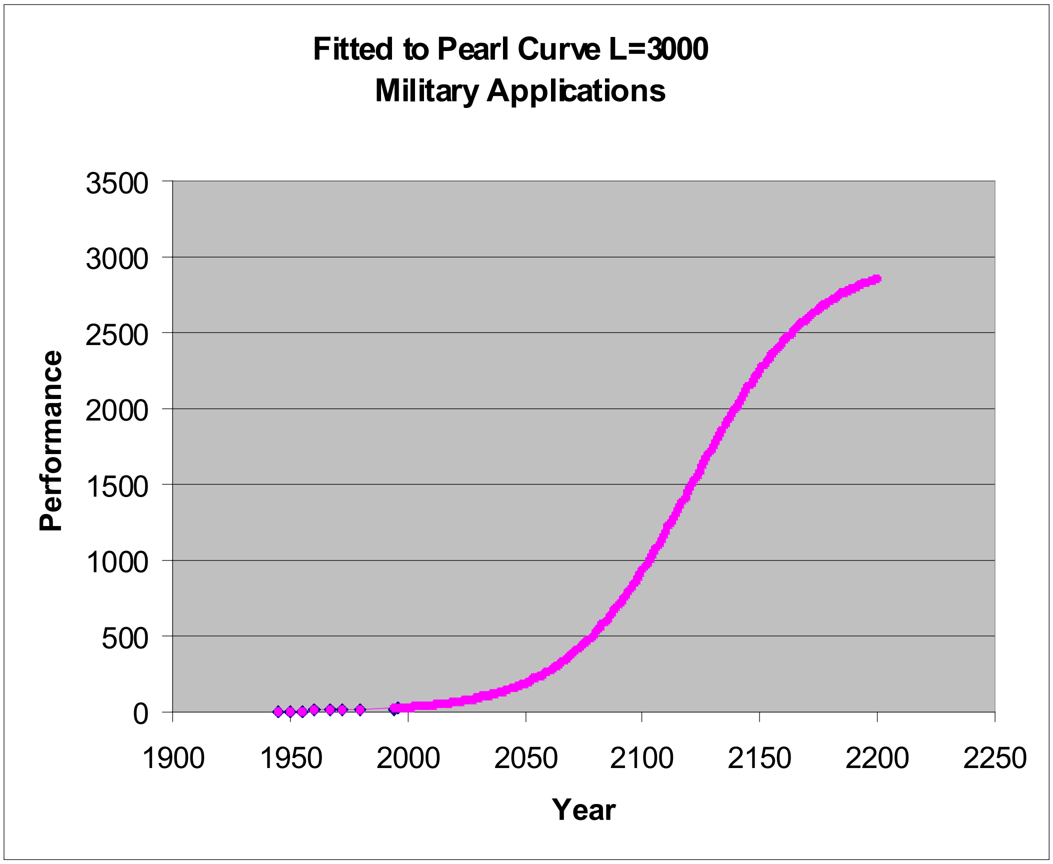
Coefficients	*L=3000*
-143.9186752	= -A
0.070778078	= B
0.85352171	Correlation

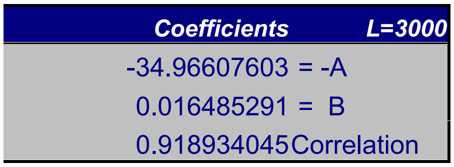
For the military applications we obtained the following data and calculated the predicted (fitted) value for “y”:

Year	Performance	y=y/(L-y)	Log10(y)	Predicted y
1945	**3**	0.0101	-1.9956	3.7282
1950	**4**	0.0135	-1.8692	4.5249
1955	**6**	0.0204	-1.6902	5.4886
1960	**8**	0.0274	-1.5623	6.6530
1967	**10**	0.0345	-1.4624	8.6970
1972	**11**	0.0381	-1.4195	10.5177
1980	**16**	0.0563	-1.2492	14.2160
1994	**16**	0.0563	-1.2492	23.8080
1996	**30**	0.1111	-0.9542	25.5874



Coefficients	*L=300*
-35.072364 = -A	
0.017055103 = B	
0.919186866Correlation	





Coefficients	L=3000
-34.96607603	= -A
0.016485291	= B
0.918934045	Correlation

6 Analysis

We must remind the reader that as with all predictions, these too are based on several assumptions, some verifiable only in retrospect. The first assumption is that context aware computing technology will exhibit growth patterns similar to the Pearl curve. Another assumption is that we selected correct representative context aware applications. A third assumption is that we have assigned those applications reasonably accurate performance rating using our somewhat naïve scale.

The raw data for each of the applications type and the calculated predictions show that military applications have a significantly different growth pace than the ones aimed for civil use. Further, assuming an identical ceiling of performance for both (regardless of the chosen ceiling), civil applications will match the military applications in their abilities circa the year 2030 and then will always exceed the military applications. For example, civil applications are predicted to reach a performance level of 200 at about 2050, while at the same time the military applications will reach at the same time a performance level of about 130. However, military related context aware computing will be a vital industry for a long time after the civil related industry will reach a stagnation point. At about 2075 civil applications of context aware computing will enter into the insignificant growth rate part of the curve, while the military applications will still have about fifty years ahead of them before reaching a similar turning point.

The fitted curve for civil applications with L=300 suggests that by 2007 context aware computing will come close to the value of “40” on our scale. By that time, we estimate that less powerful applications will have been commercialized and will have entered the business world. For example, context aware meeting rooms will become common. By 2010 context aware systems will exceed the value of “50” on our scale. Such systems will be capable of handling "softer" information and much broader concerns than the mathematical models and knowledge-based systems have been capable of handling thus far. Hence, the so-called Business Intelligence systems will be replaced or augmented by context aware applications that include consideration of much broader cultural, organizational, personal, ethical and aesthetic factors. These will be required considerations of managers using support systems, argue Mitroff and Linstone [24]. At this point in time context aware computing will become commonly integrated into mobile personal devices, as well as into the homes and cars of consumers, produced and marketed as commodities, not as a novelty.

The inflection point of a Pearl curve occurs at t=ln(a)/b, when y=L/2. Consequently, our mid-range analysis indicates that the development pace of Context Aware systems will start to slow down by 2019. However, the pace will still be considerably similar and quite rapid until 2024, by which a more noticeable slow down will occur. By the year 2028 the rate of growth will decline substantially, and the technology will reach maturity by about 2035.

The fitted curve for military applications with L=300 suggests that by 2015 context aware computing will come close to the value of "50" on our scale. One such application we envision is an autonomous reconnaissance robot for urban terrain operations. Such a device will function using very little remote supervision by human operators. Such a device could be deployed by fire fighters, by disaster recovery teams, and even operated as a scout in hostile environments by HAZMAT teams or military ground units. Response to environmental stimuli will be enabled by sensors such as cameras, microphones, sonar, infrared sensors, inclinometers, laser scanners, and micro-impulse radar, each specially tailored to a unique system architecture. Sensors customized to specific missions may also be integrated into the system. The nature or response remains to be seen, but existing technologies suggests that without ethical and legal barriers, such a robot will have the ability and authority to open fire autonomously on targets it classifies as enemy. On the other hand, such an apparatus may operate as a medical aid, providing medical stabilization to humans in areas inaccessible to other humans in a timely manner.

The inflection point of this fitted curve occurs in the year 2057. The development pace of Context Aware military systems will start to slow down at that point, but the pace will still be quite rapid until 2100, by which a noticeable slow down will occur. By the year 2125 the rate of growth will decline substantially, and the technology will reach maturity circa 2030.

7 Conclusion

The calculations and their analysis are "mechanical" and explore one aspect of growth in the area of context aware computing. Further forecast research is needed that would combine forecast in technologies that are key to implementing context aware computing. For a better assessment of these technologies a less naïve context-sensitivity objective scale needs to be developed. Human factors, such as the willingness of humans to use such systems need to be considered. Discussing usage problems relating to decision support systems, the authors of [25] write that "people have cognitive constraints in adopting intelligent systems; people do not really understand the support they get and disregard it in favor of past experience and visions; people cannot really handle large amounts of information and knowledge; people are frustrated by theories they do not really understand; and people believe they get more support by talking to other people (even if their knowledge is limited)." In this context, it is worth exploring how much common sense should Context Aware Computing possess. Should an average intelligent person have Context Aware Computing with "average" common sense, or should such a system exceed its user's mental abilities, and by how much? We must discount the other alternative, a Context Aware system that exhibits "stupidity" or low levels of "common sense", as "stupid" systems have a record of discomforting or angering their users. Forecasting the growth and key success factors of context aware computing is an area that is currently wide open and thus proposes great opportunities for further research.

Footnotes

[1] Shim JP, Warkentin M, Courtney JF, Power DJ, Sharda R, Carlsson C Past, present, and future of decision support technology Decision Support Systems volume 33 part 2 pages 111-126 June 2002

[2] C. Carlsson E. Turban in DSS: directions for the next decade, Decision Support Systems, Volume 33, Issue 2, Pages 105-110 June 2002

[3] Henry Lieberman,Ted Selker: Out of Context: Computer Systems That Adapt To, and Learn From, Context; MIT Media Laboratory http://lieber.www.media.mit.edu/people/lieber/Teaching/Context/Out-of-Context-Paper/Out-of-Context.html as of August 7, 2002

[4] Jansen, Wayne A. "Countermeasures for Mobile Agent Security." National Institute of Standards and Technology. URL: http://csrc.nist.gov/mobileagents (10 November 2000).

[5] W.H.C. Higgins, B.D. Holbrook, J.W. Emling: *Electrical Computers for Fire Control* IEEE Annals of the History of Computing, July-September 1982 (Vol. 4, No. 3) pp 218-244 available online at http://www.computer.org/annals/an1982/a3218abs.htm as of August 2002

[6] Rathert, George A , Jr Abramovitz, Marvin Gadeberg, Burnett L *The effects of powered controls and fire-control systems on tracking accuracy* NACA RM-A55D12a May 26, 1955 available on http://naca.larc.nasa.gov/reports/1955/naca-rm-a55d12a/ as of August 2002.

[7] NASA's web site http://www.dfrc.nasa.gov/History/Publications/f8ctf/foreword.html as of August 2002.

[8] Web source: http://catless.ncl.ac.uk/Risks/6.32.html#subj4 as of August 2002.

[9] Steve Stowe (First Officer and Council 16 Air Safety Chairman Delta) A Primer for Aviation Accident Investigators, Air Line Pilot, February 2000, page 18

[10] Web source: http://europa.eu.int/comm/research/rtdinf23/en/innov2.html as of August 2002.

[11] http://www.janes.com/defence/air_forces/news/jdw/jdw021108_1_n.shtml viewed Nov. 2002

[12] Want, R., Hopper, A., Falcao, V., Gibbons, J. The Active Badge Location System. ACM Transactions on Information Systems 10(1) 1992. pp. 91-102

[13] Sue Long, Dietmar Aust, Gregory D. Abowd and Chris Atkeson. Cyberguide: *Prototyping Context-Aware Mobile Applications*. CHI'96 Short paper, December 1995. http://www.cc.gatech.edu/fce/cyberguide/pubs/chi96-cyberguide.html

[14] Pascoe, J. The Stick-e Note Architecture: Extending the Interface Beyond the User. ACM International Conference on Intelligent User Interfaces, Orlando, Florida, USA 1997.. pp. 261-264

[15] Rhodes, B.J. The wearable remembrance agent: a system for augmented memory. 1st International Symposium on Wearable Computers, Cambridge, Massachusetts, October 13-14, 1997, pp.123-128.

[16] A.K. Dey, G.D. Abowd, M. Pinkerton & A. Woods. CyberDesk: A framework for providing self-integrating ubiquitous software services. Technical Report GIT-GVU-97-10, GVU Center, Georgia Tech, June 1997.

[17] Healey, J.; Picard, R.W. Startlecam: A Cybernetic Wearable Camera. 2nd. International Symposium on Wearable Computers, Pittsburgh, Pennsylvania, 19-20 October, 1998, pp.42-49.

[18] Salber, D., Dey A.K., Abowd, G.D. The Context Toolkit: Aiding the Development of Context-Enabled Applications, Proceedings of CHI'99, Pittsburgh, PA, May 15-20, 1999.

[19] Dey, A.K., Salber, D., Abowd, G.D., Futakawa, M. The Conference Assistant: Combining context-awareness with wearable computing. 3rd International Symposium on Wearable Computers, San Francisco, California, 18-19 October, 1999, pp. 21-28.

[20] Jie Yang; Weiyi Yang; Denecke, M.; Waibel, A. Smart sight: a tourist assistant system. 3rd International Symposium on Wearable Computers, San Francisco, California, 18-19 October, 1999, pp. 73-78.

[21] http://cooltown.hp.com/cooltownhome/index.asp

[22] http://research.microsoft.com/easyliving as of August 2002

[23] J. P. Shim, M. Warkentin, J. F. Courtney, D. J. Power, R. Sharda C. Carlsson: Past, present, and future of decision support technology Decision Support Systems Volume 33, Issue 2, June 2002, Pages 111-126

[24] I. Mitroff and H Linstone The Unbounded Mind: Breaking the Chains of Traditional Business Thinking, Oxford Univ. Press, New York (1993).

[25] C. Carlsson E. Turban DSS: directions for the next decade, Decision Support Systems, Volume 33, Issue 2, Pages 105-220 (June 2002)

Eli Rohn is an adjunct professor with the NJIT College of Computing Sciences, Department of Information Systems